

Jülich Supercomputing Centre (JSC)

***Laufzeitoptimierung von Simulationen
raumkontinuierlicher Modelle
der Fußgängerdynamik mithilfe
von Nachbarschaftslisten***

Jonas Mehlich

***Laufzeitoptimierung von Simulationen
raumkontinuierlicher Modelle
der Fußgängerdynamik mithilfe
von Nachbarschaftslisten***

Jonas Mehlich

Berichte des Forschungszentrums Jülich; 4308
ISSN 0944-2952
Jülich Supercomputing Centre (JSC))
Jül-4308

Vollständig frei verfügbar im Internet auf dem Jülicher Open Access Server (JUWEL) unter
<http://www.fz-juelich.de/zb/juwel>

Zu beziehen durch: Forschungszentrum Jülich GmbH · Zentralbibliothek, Verlag
D-52425 Jülich · Bundesrepublik Deutschland
☎ 02461 61-5220 · Telefax: 02461 61-6103 · e-mail: zb-publikation@fz-juelich.de

Runtime optimization of simulations based on spatially continuous models in pedestrian dynamics using neighbor list techniques

Abstract

Due to an increasing number of mass events and more complex multifunctional buildings new strategies and tools are required to guarantee the safety of pedestrians, in particular with regard to an evacuation in case of emergency. Within the project Hermes an evacuation assistant is being developed using the ESPRIT arena as a test environment. This tool is supposed to assist the decision makers (operator, police, fire department and security service) with congestion forecasts. The task of JSC is to implement a simulation of up to 60,000 pedestrians which runs faster than real-time.

Within the master's thesis this aim was supported by the use of neighbor list techniques. The optimization potential of two different techniques, Verlet and Linked-cell lists, was analyzed in test cases of homogeneous and inhomogeneous pedestrian distributions. With the aid of the neighbor list techniques a speedup factor of 70 could be achieved compared to the brute force method. However, the simulation of a basic scenario demonstrates that additional optimization techniques are required for a simulation faster than real-time.

Laufzeitoptimierung von Simulationen raumkontinuierlicher Modelle der Fußgängerdynamik mithilfe von Nachbarschaftslisten

Kurzfassung

Die steigende Anzahl von Großveranstaltungen und die wachsende Komplexität multifunktionaler Gebäude fordern neue Strategien und Hilfsmittel, um die Sicherheit der Personen, insbesondere bei einer Räumung im Gefahrenfall, zu gewährleisten. Im Rahmen des Projekts Hermes wird am Beispiel der ESPRIT arena ein Evakuierungsassistent erforscht. Dieser soll die Entscheidungsträger (Betreiber, Polizei, Feuerwehr und Sicherheitsdienste) durch Stauprognosen unterstützen. In diesem Projekt ist die Aufgabe des JSC, eine Simulation von bis zu 60.000 Fußgängern schneller als in Echtzeit zu realisieren.

Im Rahmen der Masterarbeit wurde dieses Ziel durch den Einsatz von Nachbarschaftslisten unterstützt. Zwei verschiedene Methoden, Verlet- und Linked-Cell-Listen, wurden in Testszenarien mit homogener und inhomogener Personenverteilung hinsichtlich ihres Optimierungspotentials untersucht. Mithilfe der Nachbarschaftslisten konnte im Vergleich zur Brute-Force-Methode ein Performancegewinn bis zu einem Faktor 70 erzielt werden. Die Simulation eines Beispielszenarios zeigt jedoch, dass zusätzliche Optimierungsmöglichkeiten erforderlich sind, um eine Simulation schneller als Echtzeit zu erreichen.

Inhaltsverzeichnis

1	Einleitung	1
2	Einführung in die Fußgängerdynamik	3
2.1	Wesentliche Kenngrößen	3
2.2	Kollektive Effekte	4
3	Modellierung	7
3.1	Modellierungskonzepte	7
3.2	Zelluläre Automaten	8
3.3	Kräfte-basierte Modelle	9
3.3.1	Zentrifugalkräfte-Modell	10
3.3.2	Abbildung komplexer Geometrien	13
4	Simulation raumkontinuierlicher Modelle	19
4.1	Implementierung	19
4.2	Visualisierung	22
5	Laufzeitoptimierung mithilfe von Nachbarschaftslisten	25
5.1	Grundlagen	25
5.2	Verlet-Listen	27
5.3	Linked-Cell-Listen	31
6	Performancetests	37
6.1	Bestimmung des Skin-Radius	38
6.2	Homogene Verteilung	39
6.2.1	Testszenario 1	40
6.2.2	Testszenario 2	41
6.2.3	Testszenario 3	42
6.3	Inhomogene Verteilung	44
6.4	Resultate	49
7	Anwendungsbeispiel	53

INHALTSVERZEICHNIS

8	Ausblick	57
8.1	Nachbarschaftslisten	57
8.2	Parallelisierung	58
8.3	Modellierung	62
9	Zusammenfassung	65
A	Symbole und Abkürzungen	67
A.1	Symbole	67
A.2	Abkürzungen	69

Kapitel 1

Einleitung

Die zunehmende Weltbevölkerung, wachsende Metropolen, dichter besiedelte Flächen und das verstärkte Interesse an Großveranstaltungen erfordern erhöhte Sicherheitsvorkehrungen und neue Sicherheitskonzepte, die speziell auf große Personenmengen zugeschnitten sind. In diesem Zusammenhang werden u. a. Evakuierungen und die Auswirkungen der Führung von Fluchtwegen oder der Breite von Ausgängen auf die nötige Evakuierungszeit betrachtet. Ein wichtiges Hilfsmittel zur Planung von Rettungswegen ist die Simulation von Fußgängerbewegungen. Diese wird beispielsweise bei der Konzeption von Gebäuden (z. B. Stadien, Bahnhöfe), Städten (z. B. Verkehrswege, Fußgängerzonen) und Großveranstaltungen (z. B. Konzerte, Fußballspiele, Kirchentage) oder auch zur Nachweisführung im baurechtlichen Genehmigungsverfahren eingesetzt. Für eine Simulation müssen die komplexen Bewegungsvorgänge von Fußgängern durch Modelle beschrieben werden, für die unterschiedliche Modellierungsansätze existieren. Das Ziel ist dabei, möglichst einfache Modelle zu entwickeln, die mit wenigen Parametern gesteuert werden können. Darüber hinaus führt eine geringere Komplexität der Modelle u. a. zu einer effizienteren Berechnung, was insbesondere für eine Simulation großer Personenströme in Echtzeit notwendig ist.

Ein wichtiger Anwendungsfall für eine Echtzeit-Simulation von Fußgängern ergibt sich in dem Projekt Hermes, das vom Bundesministerium für Bildung und Forschung (BMBF) gefördert wird [10]. Im Rahmen dieses Projekts wird am Beispiel der *ESPRIT arena* ein Evakuierungsassistent zur Unterstützung der Entscheidungsträger (Betreiber, Polizei, Feuerwehr und Sicherheitsdienste) erforscht. Bei einer Evakuierung sollen durch eine Simulation innerhalb einer Minute die Fußgängerbewegungen von bis zu 60 000 Besuchern während der nächsten 15 Minuten prognostiziert werden. Dies erfordert eine hohe Effizienz der durchgeführten Simulationen.

Das Funktionsprinzip des Assistenten ist in Abbildung 1.1 skizziert. Durch eine Kopplung an das Gefahrenmanagementsystem und mithilfe von Personenzählungen durch automatisierte Bildverarbeitung wird auf die aktuelle Gefahrenlage reagiert. Um die Simulation

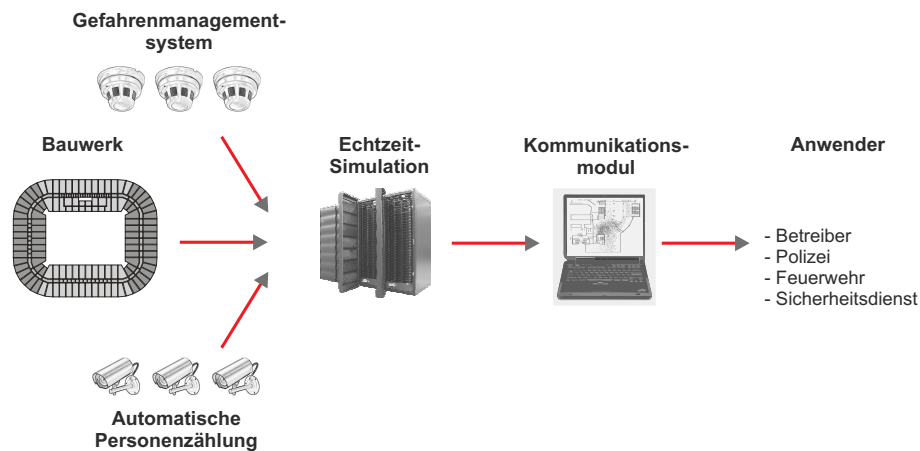


Abbildung 1.1: Prinzip des Evakuierungsassistenten, der im Projekt Hermes entwickelt wird.

schneller als in Echtzeit realisieren zu können, kommt ein Parallelcomputer zum Einsatz, der die Prognose minütlich aktualisiert. Die daraus gewonnenen Resultate werden über ein Kommunikationsmodul an die entsprechenden Entscheidungsträger weitergegeben.

Die vorliegende Arbeit ist in das Projekt Hermes integriert und verfolgt das Ziel, die Simulation schneller als in Echtzeit zu unterstützen. Anlehnend an die Verwendung im Bereich der Molekulardynamik werden zur Laufzeitorientierung Nachbarschaftslisten eingesetzt. Zwei verschiedene Techniken, Verlet- und Linked-Cell-Listen, werden hinsichtlich ihres Optimierungspotentials in verschiedenen Testszenarien untersucht.

Die Arbeit gliedert sich in folgende Abschnitte: Zu Beginn werden in Kapitel 2 Kenngrößen und Eigenschaften in der Fußgängerdynamik erläutert. Im Anschluss wird in Kapitel 3 die Modellierung von Fußgängerbewegungen thematisiert und wichtige Modellierungsansätze vorgestellt. Hierbei wird besonders auf die kräfte-basierten Modelle eingegangen, die die Interaktion von Fußgängern mit ihrer Umgebung (anderen Fußgängern, Hindernissen) mithilfe von Kräften beschreiben. Des Weiteren wird die Fußgängermodellierung im Hinblick auf eine realgetreue Abbildung komplexer Raumgeometrien untersucht. In Kapitel 4 wird das implementierte Simulationsprogramm erläutert, woraufhin das Kapitel 5 die zur Laufzeitorientierung verwendeten Nachbarschaftslisten beschreibt. Im folgenden Kapitel 6 werden die Ergebnisse der Laufzeittests der Simulationen mit den Verlet- und Linked-Cell-Listen analysiert. Daraufhin wird in Kapitel 7 die benötigte Laufzeit für eine Simulation eines Szenarios gemessen, das einen Teil der *ESPRIT arena* approximiert. In Kapitel 8 werden weitere Optimierungsmöglichkeiten genannt und verbesserte Modellierungsaspekte vorgeschlagen. Den Abschluss der Arbeit bildet Kapitel 9 mit einer Zusammenfassung.

Kapitel 2

Einführung in die Fußgängerdynamik

Im Bereich der Fußgängerdynamik existieren wichtige Kenngrößen und Effekte, anhand derer wesentliche Anforderungen an die entwickelten bzw. zu entwickelnden Modelle gestellt werden. Denn erst bei einer Übereinstimmung der experimentellen und simulierten Daten bzgl. dieser Merkmale kann davon ausgegangen werden, dass das Modell das Verhalten der Fußgänger im Wesentlichen richtig darstellt. Die Merkmale sind allerdings aufgrund des recht jungen Wissenschaftsgebiets zum Teil noch nicht ausreichend erforscht. Daher sind wissenschaftliche Experimente für die unterschiedlichsten Szenarien notwendig, um die spezifischen Kenngrößen bestimmen und entsprechend in den Modellen berücksichtigen zu können. In den nächsten Abschnitten wird auf die Kenngrößen und Effekte genauer eingegangen.

2.1 Wesentliche Kenngrößen

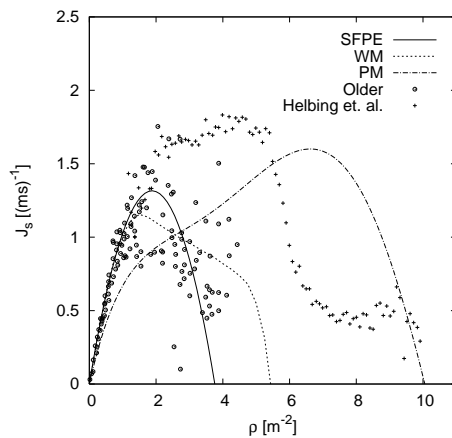
Die drei Hauptgrößen, durch die eine Menge von Personen bzw. eine einzelne Person quantitativ charakterisiert wird, sind die Geschwindigkeit v , die Dichte ρ und der Fluss J [13, 15]. Der maximale Fluss wird als Kapazität bezeichnet. Dabei ist der Fluss J definiert als die Anzahl der Personen ΔN , die einen bestimmten Ort innerhalb eines festen Zeitintervalls Δt durchqueren:

$$J = \frac{\Delta N}{\Delta t}.$$

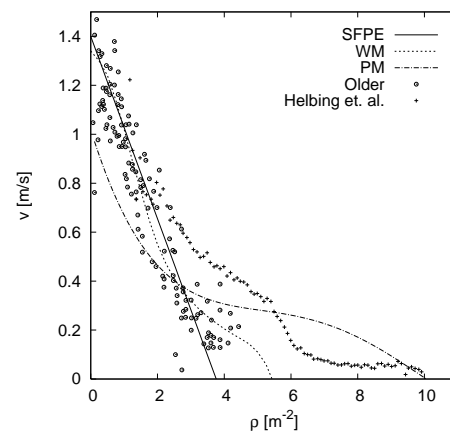
Ein anderer Ansatz stammt aus der Hydrodynamik. Dort wird der Fluss mit der Breite b der Durchflussstelle, der Durchschnittsdichte ρ und der Durchschnittsgeschwindigkeit v beschrieben:

$$J = \rho v b = J_s b.$$

Dabei ist J_s der spezifische Fluss, der den Fluss pro Einheitsbreite angibt.



(a) Beziehung zwischen Dichte und Fluss



(b) Beziehung zwischen Dichte und Geschwindigkeit

Abbildung 2.1: Fundamentaldiagramm, in dem die Relation zwischen Fluss und Dichte basierend auf verschiedenen Ergebnissen dargestellt ist. Erkennbar sind die teils deutlichen Unterschiede. (Quelle: [13])

Die Beziehung zwischen der Geschwindigkeit und der Dichte bzw. zwischen dem Fluss und der Dichte wird Fundamentaldiagramm genannt. Durch dieses Diagramm können wichtige Erkenntnisse und Aussagen über die Dynamik der Fußgänger erlangt bzw. getroffen werden. In Abbildung 2.1 werden Fundamentaldiagramme auf Basis verschiedener Messergebnisse bzgl. des Flusses und bzgl. der Geschwindigkeit gezeigt. Es wird deutlich, dass bei allen Ergebnissen die Geschwindigkeit bei steigender Dichte abfällt (2.1(a)). Es existieren jedoch im Einzelnen (teilweise extreme) Unterschiede zwischen den Ergebnissen verschiedener Untersuchungen: die Geschwindigkeit erreicht bei unterschiedlichen Dichten den Nullpunkt oder die Kapazität variiert und wird bei verschiedenen Dichten erreicht. Die Gründe der Abweichungen konnten bisher noch nicht vollständig geklärt werden. Experimente, die sowohl mit deutschen als auch mit indischen Testpersonen durchgeführt wurden, ergaben jedoch, dass z. B. der kulturelle Unterschied einen Einfluss auf die Kenngrößen hat [4].

2.2 Kollektive Effekte

In Menschenmengen auftretende Effekte werden als kollektive Phänomene bezeichnet. Hierbei handelt es sich beispielsweise um Linienbildung, die sowohl bei bidirektionalen als auch bei unidirektionalen Strömen auftritt (s. Abbildung 2.2 für einen unidirektionalen Strom). So entstehen bei einer Menge von Personen mit entgegengesetzten Bewegungsrichtungen einzelne, in der Breite variierende Strömungslinien. Weitere Effekte sind Stau an Engstellen, wie etwa Türen oder Ausgängen, oder Stauwellen, die sich entgegen der Laufrichtung der Fußgänger bewegen (insbesondere im eindimensionalen Fall [14]).

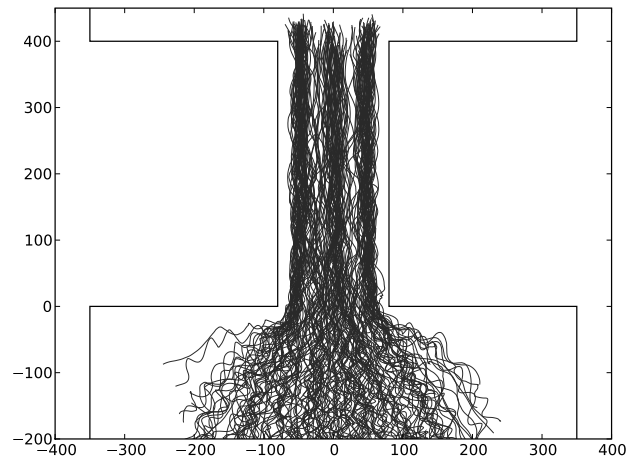


Abbildung 2.2: *Trajektorien von Fußgängern beim Durchgang einer Engstelle bei einem im Rahmen des Projekts Hermes durchgeführten Experiment. Die Linienbildung ist durch die höhere Anzahl der Trajektorien links, rechts und in der Mitte der Engstelle zu erkennen.*

Zusätzlich zu Experimenten können Informationen aus Videoaufnahmen von realen Szenarien gewonnen werden. Deren Auswertung ist allerdings aufgrund von schlechten Lichtverhältnissen oder zu geringer Bildauflösung teilweise nicht möglich. Der Vorteil bei diesem Vorgehen liegt jedoch in einem natürlichen Verhalten der Fußgänger, das bei Experimenten eventuell abweicht. Des Weiteren ist eine experimentelle Durchführung einer Notsituation (etwa eines Brands in einem Gebäude) nicht möglich, sodass im Gegensatz dazu bei vorhandenen Aufnahmen Schlüsse auf das Verhalten von Fußgängern in extremen Situationen gezogen werden können.

Kapitel 3

Modellierung

Die Modellierung des Verhaltens von Fußgängern dient dem besseren Verständnis der Fußgängerdynamik. Dabei spielt sowohl das rein wissenschaftliche Interesse eine Rolle als auch die praktische Anwendbarkeit. Im Folgenden werden unterschiedliche Modellierungskonzepte vorgestellt und zwei grundlegende Modellierungsansätze beschrieben. Hierbei werden insbesondere kräfte-basierte Modelle und im Speziellen das Zentrifugalkräfte-Modell detailliert erläutert.

3.1 Modellierungskonzepte

Es existieren unterschiedliche Modellierungskonzepte mit jeweils entgegengesetzten Eigenschaften, anhand derer die Modelle klassifiziert werden können [13]. Da auch Mischformen vorkommen, die die jeweiligen Eigenschaften kombinieren, ist eine eindeutige Klassifizierung der Modelle nicht immer möglich.

- **mikroskopisch vs. makroskopisch**

Bei mikroskopischen Modellen werden die Fußgänger als Individuen modelliert. Es ist möglich, ihnen jeweils bestimmte Eigenschaften oder Fähigkeiten zuzuweisen, aus denen eine individuelle Bewegung folgt.

Im Gegensatz dazu wird bei makroskopischen Modellen die Gesamtheit der Fußgänger beschrieben, zum Beispiel durch den Fluss oder die Dichte.

- **diskret vs. kontinuierlich**

Die Freiheitsgrade zur Beschreibung der Fußgängerbewegung sind Zeit, Raum und Zustand der Personen, etwa deren Geschwindigkeit. Jede Größe kann entweder diskret oder kontinuierlich beschrieben werden. Jegliche Kombination zwischen ausschließlich diskreten Größen und nur kontinuierlichen Größen ist möglich.

- **deterministisch vs. stochastisch**

Das Verhalten von Fußgängern kann deterministisch oder stochastisch modelliert werden. Im deterministischen Fall resultiert aus jedem Zustand ein definierter Folgezustand, wohingegen bei einer stochastischen Beschreibung bestimmte Zustandsübergänge mithilfe von Wahrscheinlichkeiten definiert sind. Häufig ergeben sich in komplexen Systemen mit der Einführung von Wahrscheinlichkeiten realistischere Ergebnisse.

- **regelbasiert vs. kräftebasiert**

In regelbasierten Modellen treffen die Fußgänger in Abhängigkeit von ihrem Zustand, ihrem Ziel und ihrer Umgebung Entscheidungen über die nächste Bewegung. Im Gegensatz dazu kann der Einfluss von anderen Personen und von der Umgebung, sowie das Streben zu einem bestimmten Ziel in Anlehnung an die Newtonsche Mechanik durch Kräfte ausgedrückt werden.

3.2 Zelluläre Automaten

Bei Modellen, die in die Kategorie der Zellulären Automaten (engl. *Cellular Automata*, CA) fallen, wird der Raum, die Zeit und der Zustand der Fußgänger diskretisiert. Die ersten Modelle mit diesem Ansatz können beispielsweise in [2], [6], und [11] gefunden werden.

Die Diskretisierung des Raums, die den Modellen den Namen verleiht, geschieht durch ein Gitter, bei dem die Zellgröße so gewählt werden kann, dass maximal eine Person eine Zelle besetzt. Beispielsweise resultiert aus einer maximalen Dichte von 6,25 Personen/m² eine Zellgröße von 40 × 40 cm², die häufig bei Zellulären Automaten verwendet wird.

Mithilfe von Regeln werden Übergangswahrscheinlichkeiten zu einer Nachbarzelle definiert. Diese setzen sich aus der gewünschten Bewegungsrichtung, der Interaktion zu anderen Fußgängern und der Interaktion zur Umgebung (Hindernisse, Wände, etc.) zusammen. Dabei kann die Nachbarschaft zum Beispiel als Von-Neumann- oder als Moore-Nachbarschaft festgelegt werden (s. Abb. 3.1(a) bzw. 3.1(b)). Im einfachsten Fall bewegt sich eine Person maximal eine Zelle pro Zeitschritt weiter. Durch eine größere und unterschiedliche Anzahl von Zellen, die pro Zeitschritt durchquert werden dürfen, können verschiedene Geschwindigkeiten realisiert werden.

Zelluläre Automaten zeichnen sich durch ihre einfachen Regeln und ihre kurzen Rechenzeiten aus. Im Gegensatz zu kräftebasierten, raumkontinuierlichen Modellen müssen keine Differentialgleichungen gelöst werden und auch die Zeitschritte sind ca. um den Faktor 10 bis 100 größer, da sie sich nach der Zellgröße richten. Ein weiterer Vorteil bezüglich der Rechenzeit ist die durch die Diskretisierung mögliche Verwendung von Integer-Arithmetik.

Die Raumdiskretisierung birgt jedoch auch Nachteile: Durch sie entstehen etwa bei der

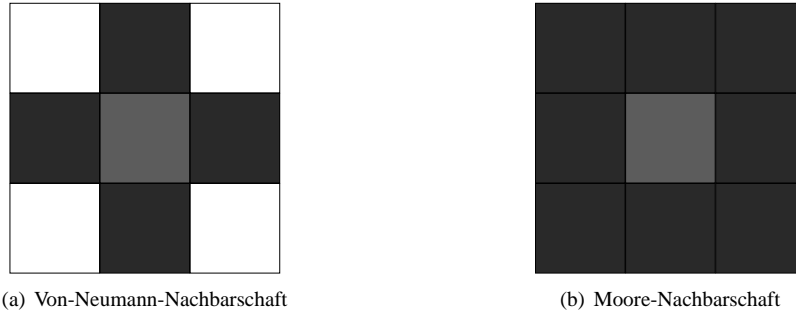


Abbildung 3.1: Mögliche Nachbarschaften: horizontale und vertikale Nachbarzellen bei der Von-Neumann-Nachbarschaft und alle 8 Nachbarzellen bei der Moore-Nachbarschaft.

Abbildung der Umgebung Diskretisierungsfehler, da eine Repräsentation einer Geometrie nur approximiert werden kann. Durch kleinere Zellen kann dem entgegengewirkt werden. Die räumliche Diskretisierung kann durch die fest vorgegebenen Intervalle zudem bei der Bewegung der Fußgänger für Probleme sorgen, insbesondere bei Bewegungen, die nicht in Richtung der Zellaufteilung, sondern diagonal verlaufen.

3.3 Kräfte-basierte Modelle

Bei den raumkontinuierlichen, mikroskopischen kräfte-basierten Modellen wird das Verhalten von Fußgängern so interpretiert, dass Kräfte ihrer Umgebung auf sie einwirken. Durch die Kräfte erfahren die Fußgänger eine Beschleunigung, sodass eine gewisse Analogie zur klassischen Mechanik besteht, und die Bewegung eines i -ten Fußgängers demnach mathematisch durch ein System von Differentialgleichungen beschrieben werden kann:

$$m_i \frac{d}{dt} \vec{v}_i = \vec{F}_i = \vec{F}_i^{drv} + \vec{F}_i^{rep} \quad \text{und} \quad \frac{d}{dt} \vec{x}_i = \vec{v}_i. \quad (3.1)$$

Dabei ist m_i die Masse, \vec{x}_i die Position und \vec{v}_i die Geschwindigkeit des i -ten Fußgängers. In den Simulationen wurde $m_i = 1$ gewählt.

\vec{F}_i^{drv} ist die treibende Kraft (engl. *driving*), die auf einen Fußgänger wirkt, der zu einem bestimmten Ziel gelangen möchte, und wird durch die Wunschrichtung und -geschwindigkeit ausgedrückt. Die treibende Kraft wird häufig durch

$$\vec{F}_i^{drv} = m_i \frac{\vec{v}_i^0 - \vec{v}_i}{\tau} \quad (3.2)$$

definiert, wobei \vec{v}_i^0 die jeweilige Wunschrichtung und -geschwindigkeit und τ ein Relaxationskoeffizient zur Steuerung der Beschleunigung ist [8].

\vec{F}_i^{rep} beschreibt die abstoßende Kraft (engl. *repulsive*), die sich durch das Vermeiden einer zu geringen Nähe zu anderen Personen oder Hindernissen bzw. Wänden ergibt. Für den

i -ten Fußgänger gilt demnach:

$$\vec{F}_i^{rep} = \sum_{i \neq j} \vec{f}_{ij}^{rep} + \sum_w \vec{f}_{iw}^{rep},$$

wobei \vec{f}_{ij}^{rep} die abstoßende Kraft zur j -ten Person und \vec{f}_{iw}^{rep} die zum w -ten Hindernis ist.

Die Schwierigkeit bei kräfte-basierten Modellen ist die konkrete Formulierung der einzelnen Kräfte. Während die treibende Kraft weitestgehend bekannt ist und wie in Gleichung (3.2) definiert wird, existieren für die abstoßenden Kräfte verschiedene Definitionen [8, 12, 17, 21]. Sie können etwa durch eine Exponentialfunktion oder eine gebrochen-rationale Funktion dargestellt werden, bei der sich die Kraft umgekehrt proportional zum Abstand verhält, d. h. bei geringerem Abstand zu einer anderen Person oder zu einer Wand ist die wirkende Kraft größer und umgekehrt. Häufig funktioniert das Zusammenspiel der treibenden und abstoßenden Kräfte nur bedingt, sodass es zu Durchdringungen der Fußgänger oder zu hohen Kräften und dadurch zu (deutlich) höheren Geschwindigkeiten bzw. auch zu negativen Geschwindigkeiten oder Oszillationen kommt. Gelöst werden diese Probleme meist mithilfe einer Konfliktlösung, die jedoch die Gesamtdynamik des Systems stark beeinflusst. Es gibt Ansätze, die Konfliktlösung zu vermeiden und ein nur auf Kräfte basierendes Modell zu entwickeln [5].

3.3.1 Zentrifugalkräfte-Modell

Bei dem Zentrifugalkräfte-Modell (engl. *Centrifugal Force Model, CFM*) nach [21] sollen in die Formulierung der abstoßenden Kräfte zwei Eigenschaften einfließen. Zum einen hängt die Krafteinwirkung zwischen zwei Personen von deren Abstand und ihrer relativen Geschwindigkeit ab, zum anderen wird der Blickwinkel einer Person berücksichtigt.

Die relative Geschwindigkeit v_{ij} wird mithilfe des Skalarprodukts ($\langle \cdot, \cdot \rangle$) berechnet:

$$v_{ij} = \frac{1}{2} (\langle \vec{v}_i - \vec{v}_j, \vec{e}_{ij} \rangle + |\langle \vec{v}_i - \vec{v}_j, \vec{e}_{ij} \rangle|), \quad (3.3)$$

$$\text{mit } \vec{e}_{ij} = \frac{\vec{R}_{ij}}{\|\vec{R}_{ij}\|}$$

$$\text{und } \vec{R}_{ij} = \vec{x}_i - \vec{x}_j. \quad (3.4)$$

Dabei ist \vec{e}_{ij} der Richtungsvektor von der i -ten zur j -ten Person und \vec{R}_{ij} der Differenzvektor zwischen beiden Personen an Position $\vec{x}_i = (x_i, y_i)^T$ bzw. $\vec{x}_j = (x_j, y_j)^T$, wie in Abbildung 3.2 skizziert. Durch die Definition in Gleichung (3.3) wirkt bei einer positiven relativen Geschwindigkeit eine Kraft, bei einer negativen relativen Geschwindigkeit hingegen nicht.

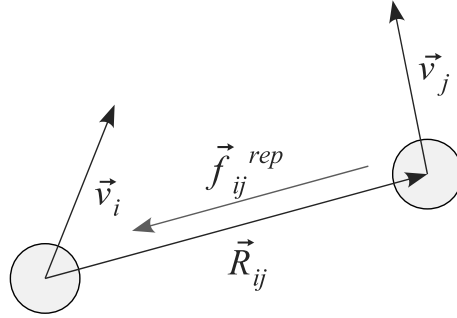


Abbildung 3.2: Für die abstoßende Kraft \vec{f}_{ij}^{rep} zwischen zwei Fußgängern wird u. a. die relative Geschwindigkeit verwendet, die mithilfe des Differenzvektors \vec{R}_{ij} und der Geschwindigkeitsvektoren der beiden Fußgänger \vec{v}_i und \vec{v}_j berechnet wird.

Die Berücksichtigung des Blickwinkels wird mit einem Koeffizienten K_{ij} eingeführt:

$$K_{ij} = \frac{1}{2} \left(\frac{\langle \vec{v}_i, \vec{e}_{ij} \rangle + |\langle \vec{v}_i, \vec{e}_{ij} \rangle|}{\|\vec{v}_i\|} \right), \quad K_{ij} = 0 \quad \text{wenn} \quad \|\vec{v}_i\| = 0 \quad (3.5)$$

Dadurch erhalten nur die Kräfte derjenigen Personen Einfluss, die sich in einem Blickwinkel von $\pm 90^\circ$ bzgl. der Bewegungsrichtung der i -ten Person befinden.

Mit den in den Gleichungen (3.3) und (3.5) definierten Größen erhält die interaktive Kraft zwischen zwei Fußgängern folgende Form:

$$\vec{f}_{ij}^{rep} = -m_i K_{ij} \frac{v_{ij}^2}{\|\vec{R}_{ij}\|} \vec{e}_{ij}. \quad (3.6)$$

Die gleiche Kraft gilt auch für die Abstoßung zu Hindernissen bzw. Wänden:

$$\vec{f}_{iw}^{rep} = -m_i K_{iw} \frac{v_{iw}^2}{\|\vec{R}_{iw}\|} \vec{e}_{iw}. \quad (3.7)$$

Da Hindernisse keine eigene Geschwindigkeit besitzen, handelt es sich in diesem Fall bei der relativen Geschwindigkeit nur um die Geschwindigkeit des i -ten Fußgängers.

Die treibende Kraft wird wie in Kapitel 3.3 definiert und lautet:

$$\vec{F}_i^{drv} = m_i \frac{\vec{v}_i^0 - \vec{v}_i}{\tau}. \quad (3.8)$$

Insgesamt ergibt sich für das CFM mit den Definitionen in Gleichung (3.6), (3.7) und (3.8) die auf die i -te Person wirkende Kraft

$$\vec{F}_i = \vec{F}_i^{drv} + \vec{F}_i^{rep} = \vec{F}_i^{drv} + \sum_{i \neq j} \vec{f}_{ij}^{rep} + \sum_w \vec{f}_{iw}^{rep}. \quad (3.9)$$

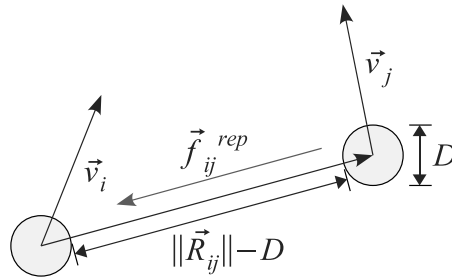


Abbildung 3.3: Für die abstoßende Kraft \vec{f}_{ij}^{rep} zwischen zwei Fußgängern wird u. a. die relative Geschwindigkeit verwendet. Dabei muss das von den Fußgängern eingenommene Volumen berücksichtigt werden, sodass sich ein Abstand von $\|\vec{R}_{ij}\| - D$ ergibt.

Mit Gleichung (3.1) wird die Bewegung träger Teilchen beschrieben, sodass nicht gewährleistet ist, dass Durchdringungen auftreten und dadurch ein unrealistisches Verhalten von Fußgängern entsteht. In [21] wurde daher zur Konfliktlösung eine Kollisionserkennung (engl. *Collision Detection Technique*, CDT) eingesetzt. Dabei wird nach der Iteration des nächsten Zeitschrittes überprüft, ob Kollisionen entstanden sind. Die entsprechenden Personen werden anschließend auf die alte Position gesetzt bzw. in eine Richtung mit einem bestimmten Winkel bewegt, der von der aktuellen Bewegungsrichtung abhängig ist. Die Konfliktlösung erlaubt große Zeitschritte von 0,05 s bis 0,15 s, ohne dass es zu signifikanten Unterschieden kommt. Allerdings beeinträchtigt der Gebrauch einer Konfliktlösung die Dynamik des Systems. Daher wurden in [5] Änderungen vorgeschlagen, die ein rein auf Kräfte basierendes Modell ermöglichen, auf die im folgenden Abschnitt eingegangen wird.

Modifikationen

Durch den in Gleichung (3.4) definierten Abstand zwischen zwei Fußgängern bzw. zwischen einem Fußgänger und einem Hindernis entsteht in den abstoßenden Kräften (Gleichungen (3.6) und (3.7)) eine Singularität bei $\|\vec{R}_{ij}\| = 0$, die zu einer hohen Kraft bei geringen Abständen führt. Die Krafteinwirkung ist jedoch erst dann ausreichend hoch, wenn der Abstand so klein ist, dass sich die Personen überlappen würden. Um die unrealistische Überlappung zu vermeiden, wird die Singularität verschoben. Wird der Platzbedarf der Fußgänger in Form von Scheiben bzw. Kreisen mit Durchmesser D dargestellt (siehe Abbildung 3.3), ergeben sich aus den Gleichungen (3.6) und (3.7) folgende neue Definitionen, die einen Volumenausschluss garantieren:

$$\vec{f}_{ij}^{rep} = -m_i K_{ij} \frac{v_{ij}^2}{\|\vec{R}_{ij}\| - D} \vec{e}_{ij} \quad (3.10)$$

und

$$\vec{f}_{iw}^{rep} = -m_i K_{iw} \frac{v_{iw}^2}{\|\vec{R}_{iw}\| - D} \vec{e}_{iw}. \quad (3.11)$$

Weiterhin wurde in [16] festgestellt, dass der von den Fußgängern benötigte Platzbedarf abhängig von deren Geschwindigkeit ist, denn bei höherer Geschwindigkeit ist die Schrittweite größer. Für die i -te Person gilt der Zusammenhang:

$$D_i = D_i(\vec{v}_i) = D_{min} + D_t \|\vec{v}_i\|, \quad (3.12)$$

wobei D_{min} und D_t frei wählbare Parameter sind. In den Simulationen wurde $D_{min} = 0,2$ m und $D_t = 0,2$ gewählt.

Mit einer weiteren Modifikation und der Definition in Gleichung (3.12) resultiert aus den Gleichungen (3.10) und (3.11) die endgültige Form der abstoßenden Kräfte:

$$\vec{f}_{ij}^{rep} = -m_i K_{ij} \frac{(v_{ij} + \alpha \|\vec{v}_i^0\|)^2}{\|\vec{R}_{ij}\| - \frac{1}{2}(D_i + D_j)} \vec{e}_{ij}, \quad (3.13)$$

und

$$\vec{f}_{iw}^{rep} = -m_i K_{iw} \frac{(v_{iw} + \beta \|\vec{v}_i^0\|)^2}{\|\vec{R}_{iw}\| - \frac{1}{2}D_i} \vec{e}_{iw}, \quad (3.14)$$

wobei α und β dazu dienen, die Stärke der Kraft zu variieren. In den Simulation wurde $\alpha = \beta = 0,28$ gesetzt.

Im Folgenden wird $\|\vec{R}_{ij}\|$ als Abstand und

$$E = E(D_i, D_j) = \|\vec{R}_{ij}\| - \frac{1}{2}(D_i + D_j) \quad (3.15)$$

als effektiver oder Interaktions-Abstand bezeichnet.

Bei Simulationen unter Verwendung dieser abstoßenden Kräfte kommt es zum Teil besonders bei großen Zeitschritten zu Durchdringungen der Fußgänger oder extrem hohen Geschwindigkeiten. Durch eine Interpolation der Kraftfunktion im Bereich der Singularität werden unendlich große Kräfte vermieden. Des Weiteren wird aufgrund der kurzreichweitigen Kräftewirkung, bei der die Kraft ab einer bestimmten Distanz Null ist, die Kraftfunktion ab einem effektiven Abstand E_{max} auf Null interpoliert, um eine Unstetigkeit zu vermeiden. Dieser Abstand wird hier auf 2,5 m gesetzt. Mithilfe der Interpolationen treten bei größeren Zeitintervallen keine Durchdringungen oder zu hohe Geschwindigkeiten mehr auf.

3.3.2 Abbildung komplexer Geometrien

Bei der Modellierung stellt sich insbesondere im Hinblick auf spätere Simulationen realer Szenarien die Frage, wie eine Interaktion der Fußgänger mit einer komplexen Geometrie

zu realisieren ist. Dies ist jedoch auch zur Nachbildung von Experimenten zur Validierung des Modells anhand der empirischen Daten relevant. Bisher wurde nur die Darstellung der Fußgänger in Form von Kreisen konkreter besprochen, jedoch nicht, wie beispielsweise Räume, Wände und Ausgänge abgebildet werden können. Des Weiteren ist nicht definiert, welche Wand oder welcher Punkt einer Wand oder eines Ausgangs für die Kraftberechnung verwendet wird.

Zur Modellierung des Fußgängerverhaltens innerhalb einer komplexen Raumstruktur ist zu klären,

- wie ein einzelner Raum realisiert werden kann,
- wie mehrere Räume zu komplexen Geometrien verbunden werden können,
- wie die Wunsch- bzw. Zielrichtung und damit die treibende Kraft der Fußgänger bestimmt werden kann.

Modellierung eines Raumes

Es werden einfachheitshalber nur rechtwinklige Räume simuliert, deren Wände¹ parallel zu den Koordinatenachsen liegen. Zur Berechnung der abstoßenden Kraft wird der effektive Abstand eines Fußgängers zu den Wänden des Raumes genutzt, sodass die Lage der Wände bekannt sein muss. Ein Rechteck zur Repräsentation eines Raumes kann über vier Koordinatenpaare (Ecken) dargestellt werden oder durch die Angabe der linken, rechten, oberen und unteren Position der Seiten.

Modellierung komplexer Geometrien

Komplexe Geometriestrukturen werden durch die Verknüpfung von Räumen durch Türen bzw. Ausgängen erzeugt. Das daraus resultierende Räume-Netzwerk kann durch eine Graphendarstellung repräsentiert werden. Dabei entsprechen die Räume den Knoten und die Türen den Kanten des Graphen. In einer Raum-Ausgangs-Matrix können ähnlich einer Inzidenzmatrix² zu den jeweiligen Räumen die entsprechenden (offenen) Ausgänge erfasst werden.

Modellierung von Ausgängen und die Problematik der Zielbestimmung

Ein Ausgang wird durch die Angabe zweier Koordinatenpaare (e_x^1, e_y^1) und (e_x^2, e_y^2) bestimmt, wobei das erste Paar den oberen bzw. linken Eckpunkt und das zweite Paar

¹Die Wände haben in diesem Fall keine Ausdehnung. Eine Ausdehnung hätte eventuell nur Einfluss bei den Ausgängen, wird hier jedoch zur Simplifizierung vernachlässigt.

²In der Graphentheorie kann zur Darstellung eines Graphen mit n Knoten und m Kanten eine $n \times m$ -dimensionierte Inzidenzmatrix verwendet werden. Dabei werden in jeder der m Spalten die zwei durch die jeweilige Kante verbundenen Knoten markiert.

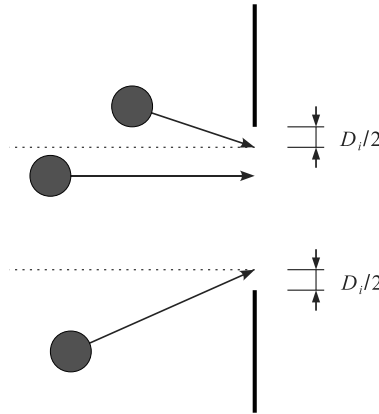


Abbildung 3.4: Ein Fußgänger geht auf dem kürzesten Weg zu seinem Ziel.

den unteren bzw. rechten Eckpunkt markiert. Aufgrund der Lage der Räume sind die Ausgänge entweder horizontal oder vertikal ausgerichtet.

Die Probleme der Zielbestimmung werden anhand eines vertikal ausgerichteten Ausganges erläutert. Dieselben Überlegungen treffen analog auch auf horizontale Ausgänge zu.

Üblicherweise bewegt sich ein Fußgänger auf dem kürzesten Weg zu seinem Ziel. Das bedeutet, es wird der direkte Weg zu den Ecken des Ausganges gewählt, wenn sich der Fußgänger über- oder unterhalb des Ausganges befindet, ansonsten der direkte Weg (senkrecht) zum Ausgang. Bei der Zielbestimmung muss jedoch der jeweils benötigte Platzbedarf der Fußgänger berücksichtigt werden, wie in Abbildung 3.4 dargestellt. Daher ist der Zielpunkt für einen Fußgänger, der sich ober- bzw. unterhalb eines Ausganges befindet, nicht die jeweilige Ecke des Ausganges ((e_x^1, e_y^1) bzw. (e_x^2, e_y^2)), sondern ein um $D_i/2$ verschobener Punkt, d. h. $(e_x^1, e_y^1 - D_i/2)$ bzw. $(e_x^2, e_y^2 + D_i/2)$.

Simulationen zeigen jedoch, dass es durch diesen Ansatz zu Blockierungen an den Ecken kommt und der Platz in der Mitte nicht genutzt wird. Eine bessere Dynamik und Platznutzung des mittleren Bereiches werden durch einen Zielpunkt erreicht, der sich in der Mitte des Ausganges befindet, es sei denn, der Fußgänger befindet sich in dem Bereich $e_y^1 - D_i/2 \leq y_i \leq e_y^2 + D_i/2$. Innerhalb dieses Bereichs kann der Fußgänger direkt auf den Ausgang zugehen, wie in Abbildung 3.5 verdeutlicht wird.

Sobald ein Fußgänger einen Raum verlässt und einen neuen Raum betritt, ist der nächstliegende Ausgang dieses Raums das neue Ziel. Dabei darf jedoch nicht der soeben durchquerte Ausgang benutzt werden, ansonsten würden die Fußgänger wieder umkehren. Um dies zu verhindern, darf in der oben beschriebenen Raum-Ausgangs-Matrix jeder Ausgang nur von einem Raum als offen markiert werden. Die Darstellung kann somit als eine Markierung von Fluchtwegen interpretiert werden.

Bislang findet eine Berechnung der interaktiven Kräfte zwischen den Fußgängern ohne Beachtung der Abschirmung durch Wände statt. Hierbei muss jedoch berücksichtigt

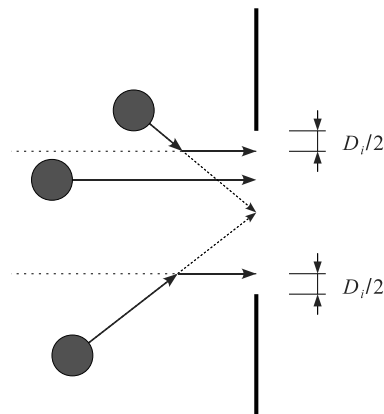


Abbildung 3.5: Eine bessere Dynamik wird dadurch erzeugt, dass die Fußgänger zur Mitte des Ausgangs gehen, wenn sie sich ober- oder unterhalb des Ausgangs befinden. Ansonsten wird der direkte Weg gewählt.

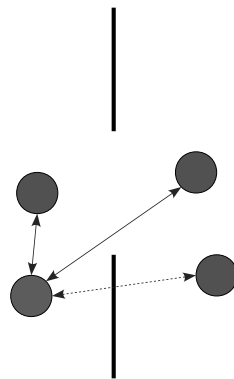


Abbildung 3.6: Eine Berechnung von Wechselwirkungen zwischen Personen in unterschiedlichen Räumen muss erfolgen, wenn Sichtkontakt besteht (durchgezogene Linie). Anderenfalls wird keine Kraft berechnet (gestrichelte Linie).

werden, dass eine interaktive Wechselwirkung zwischen Personen durch Wände hindurch in der Realität nicht existiert und daher vermieden werden muss. Generell können diese Kraftwirkungen verhindert werden, indem überprüft wird, in welchen Räumen sich die beteiligten Fußgänger befinden. Bei Personen in unterschiedlichen Räumen wird demzufolge die wechselwirkende Kraft ignoriert. Im Bereich der Ausgänge können sich die Personen jedoch in unterschiedlichen Räumen aufhalten, aber dennoch eine Kraft aufeinander ausüben. Dies ist, wie in Abbildung 3.6 veranschaulicht, bei Sichtkontakt gegeben. Daher muss bei der Kraftberechnung zum einen der Raum in Betracht gezogen werden und zum anderen eine Überprüfung auf Sichtkontakt erfolgen.

In dem Modell, das im Rahmen dieser Arbeit verwendet wird, werden diese Aspekte nicht berücksichtigt. Die Fehler durch diese Vereinfachung sind zu vernachlässigen, da sie die späteren Testszenarien zur Laufzeitmessung nicht oder kaum beeinflussen. Für den Testfall einer homogenen Verteilung (s. Kapitel 6.2) ergibt sich das Problem einer Kraftwech-

selwirkung durch Wände hindurch nicht, da alle Personen innerhalb eines Raums bleiben. Im inhomogenen Testfall (s. Kapitel 6.3) wird die Krafteinwirkung sowohl bezüglich der Laufzeit als auch bezüglich der Dynamik des Fußgängerverhaltens kaum eine Rolle spielen, da zum einen die Anzahl der „falschen“ Kraftberechnungen gering ist und zum anderen die jeweiligen Kräfte nahezu oder gleich Null sind. Beides ist damit zu begründen, dass sich vor einem Ausgang ein Stau bildet und daher die Geschwindigkeiten der Fußgänger dort geringer sind als die der Personen, die sich hinter dem Ausgang befinden und sich mit nahezu Wunschgeschwindigkeit fortbewegen können. Daher ist die relative Geschwindigkeit $v_{ij} = 0$ und demzufolge die wirkende Kraft Null (s. Gleichung (3.3) in Kapitel 3.3.1). Des Weiteren verlassen die Personen hinter dem Ausgang aufgrund der hohen Geschwindigkeit in kurzer Zeit den relevanten Einflussbereich der Personen des anderen Raums und tragen somit nur geringfügig zur Kraftberechnung bei.

Kapitel 4

Simulation raumkontinuierlicher Modelle

Im Folgenden werden Implementierungsdetails des Programms erläutert, das zur Simulation verwendet wird. Dabei werden sowohl Berücksichtigungen bezüglich der Realisierung auf einem Cell-Prozessor angesprochen als auch der generelle Ablauf des Programms beschrieben. Weiterhin wird auf die Rolle der Visualisierung der Simulationsergebnisse eingegangen.

4.1 Implementierung

Im Rahmen dieser Arbeit wurde mit der Programmiersprache C++ ein Programm entwickelt, das der Simulation raumkontinuierlicher Fußgängermodelle dient. In diesem Fall wurde das in Kapitel 3.3.1 beschriebene Zentrifugalkräfte-Modell mit den dort aufgeführten Modifikationen implementiert. Durch den modularen Programmaufbau können jedoch weitere Modelle integriert werden. Die Programmstruktur wurde so entworfen, dass eine spätere Parallelisierung auf einem Cell-Prozessor (s. Kapitel 8.2) effizient möglich ist. Dabei mussten Kompromisse zwischen reiner Objektorientierung und Effizienz (bzgl. des Cell-Prozessors) gemacht werden. Beispielsweise werden zur höheren Performance die Personen nicht einzeln in Objekten implementiert, sondern die beschreibenden Daten aller Personen (Position, Geschwindigkeit, Raumnummer, etc.) in einem Array gespeichert, was Vorteile bei der Speicherverwaltung auf dem Cell-Prozessor bringen soll. Erwähnenswert ist zudem, dass generell nicht mit doppelt genauen Gleitkommazahlen (`double`), sondern mit einfach genauen Gleitkommazahlen (`float`) gerechnet wird, da der Cell-Prozessor für Letztere optimiert ist und weniger Speicher benötigt wird.

Zum Lösen der Differentialgleichung wird das Euler-Verfahren mit einem Zeitschritt $\Delta t = 0,01s$ verwendet. Ein Vergleich zum Runge-Kutta-Verfahren vierter Ordnung zeigt

Initialisierung
Schleife über alle Zeitschritte, t aktueller Zeitschritt
Nachbarschaftslisten aktualisieren
Differentialgleichung für Zeitschritt t lösen
Raum und Ausgang für jeden Fußgänger aktualisieren
Daten des aktuellen Zeitschritts t in eine Datei schreiben

Abbildung 4.1: Hauptroutine des Simulationsprogramms.

te keine relevanten Unterschiede. Hinzu kommt, dass neben der ohnehin schon erhöhten Rechenintensität des Runge-Kutta-Verfahrens weitere Probleme bei der Verwendung von Linked-Cell-Listen bestehen (s. Kapitel 5.3), die zu noch höheren Rechenzeiten führen würden.

In Abbildung 4.1 ist der Ablauf der Hauptroutine des Simulationsprogramms dargestellt. Zunächst werden alle nötigen Daten initialisiert. Dabei werden u. a. die Fußgänger in einer frei wählbaren Region, meist innerhalb eines Raums, zufällig positioniert (siehe unten) und Raum- und Ausgangsinformation gesetzt. In einer Schleife über eine gewünschte Zeitdauer wird das System von Differentialgleichungen numerisch gelöst und dadurch die neue Position und Geschwindigkeit der Fußgänger bestimmt. In jedem Zeitschritt werden der Raum und der nächstgelegene Ausgang gesetzt. Anschließend werden die relevanten Daten (meist die Positionen der Fußgänger) für eine spätere Analyse bzw. Visualisierung in eine Datei geschrieben. Je nach Verwendung von Nachbarschaftslisten erfolgt in jedem Zeitschritt oder auch nur in einigen Zeitschritten eine Aktualisierung der Listen (s. Kapitel 5.2 bzw. 5.3 für eine genaue Beschreibung).

Für eine Simulation werden die Personen in einem bestimmten Startbereich zufällig positioniert. Dabei müssen Überlappungen der Personen und Durchdringungen mit Wänden vermieden werden. Zum Konstruieren einer ungeordneten Verteilung kann der entsprechende Bereich in ein Gitter mit quadratischen Feldern der Kantenlänge $D_{min} + 0,05 \text{ m} = 0,25 \text{ m}$ unterteilt werden (s. Kapitel 3.3.1) und eine Person innerhalb jedes zweiten Feldes zufällig positioniert werden (s. Abbildung 4.2). Dabei muss das Gitter zu den Wänden einen ausreichenden Abstand einhalten. Diese Methode lässt keine Dichten größer als 4 Personen/m^2 zu. Da vor Engstellen jedoch höhere Dichten auftreten und daher ein Laufzeitverhalten auch bei höheren Dichten untersucht werden soll, wird ein anderer Ansatz zur Positionierung der Fußgänger verwendet.

Für höhere Dichten und eine einfache zu realisierende Verteilung wird ein regelmäßiges

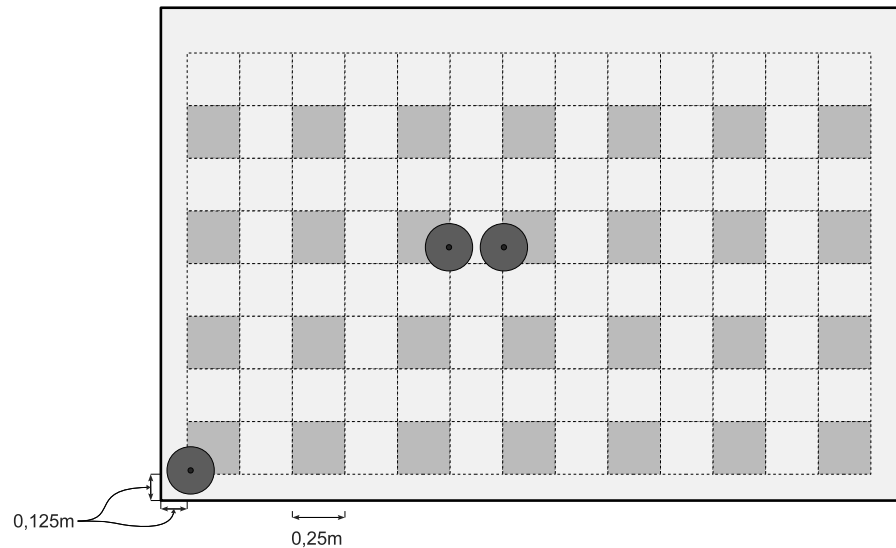
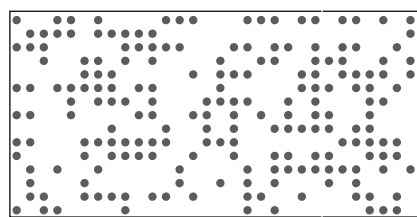
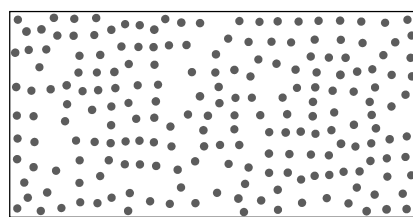


Abbildung 4.2: Zur zufälligen Positionierung der Fußgänger wird der Raum in quadratische Felder mit Kantenlänge D_{\min} aufgeteilt. Die möglichen zu besetzenden Felder sind blau markiert. Mögliche Positionierungen von einigen Fußgängern veranschaulichen das Prinzip der Verteilung.



(a) Positionen der Fußgänger bei $t = 0$ s



(b) Positionen der Fußgänger bei $t = 10$ s

Abbildung 4.3: Die zweite Verteilungsmöglichkeit ergibt eine teils regelmäßige Struktur (4.3(a)), die jedoch nach bereits 10 Sekunden keine Ordnung mehr aufweist (4.3(b)). Als Beispiel dient der homogene Testfall (s. Kapitel 6.2), gleiches gilt für das inhomogene Szenario (s. Kapitel 6.3).

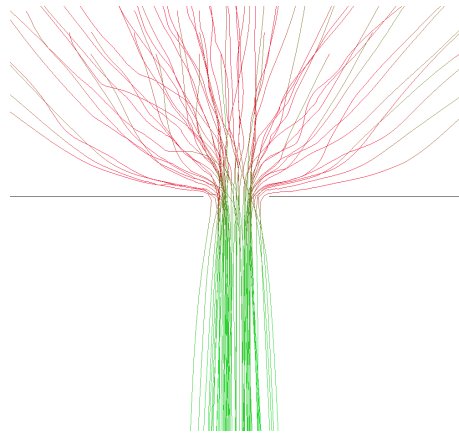


Abbildung 4.4: Visualisierung von Trajektorien bei einer Engstelle mit dynamischer Farbwahl: rot symbolisiert geringe und grün hohe Geschwindigkeiten.

Gitter verwendet. Die Gitterbreite entspricht 33 cm, sodass bei voller Besetzung des Gitters eine maximale Dichte von ca. 9 Personen/m² erreicht werden kann. Bei niedrigeren Dichten bleiben Gitterpunkte unbesetzt, wodurch eine weniger regelmäßige Struktur entsteht. Nach kurzen Simulationszeiten ergibt sich aus der regelmäßigen Anordnung eine unregelmäßige Struktur, wie Abbildung 4.3 zeigt, sodass diese Verteilungsart die Dynamik der Fußgänger im Wesentlichen nicht beeinflusst. Daher wird diese Methode zur Anfangspositionierung für die späteren Testfälle eingesetzt.

4.2 Visualisierung

Da es sich im Bereich der Fußgängerdynamik um dynamische Systeme handelt, hat die Visualisierung insbesondere in Form von Animationen eine große Bedeutung. Zum einen dient sie der Darstellung der experimentellen und simulierten Ergebnisse, um dynamische Effekte, beispielsweise Stauphänomene, beobachten zu können. Des Weiteren unterstützt eine Visualisierung die Modellentwicklung und ist ein wichtiges Hilfsmittel für die Fehleranalyse bei der Implementierung.

Im Rahmen dieser Arbeit wurde in C++ mit *OpenGL (Open Graphics Library)* ein Programm zur Visualisierung von Trajektoriendaten entwickelt. Zur Darstellung werden sowohl die Geometrieinformationen als auch die Fußgängerdaten eingelesen, die zuvor durch das Simulationsprogramm erzeugt wurden. Das Programm erlaubt unterschiedliche Visualisierungsmöglichkeiten: alle oder ausgewählte Fußgänger in Form von Kreisen und die (jeweiligen) Trajektorien mit Linien oder Punkten in unterschiedlichen Farben. Zur Verdeutlichung der Möglichkeiten zeigt Abbildung 4.4 die Trajektorien einer Simulation im Bereich einer Engstelle, wobei die Farbe die Geschwindigkeit der Personen repräsentiert. Rot deutet auf eine langsame Geschwindigkeit hin, wohingegen grün eine

schnellere Bewegung betont. Zur besseren Übersicht ist es zudem möglich, alle Daten eines bestimmten Zeitintervalls anzuzeigen. Des Weiteren können zwei Datensätze zeitgleich visualisiert werden. Diese Funktionalität ermöglichte es, während der Programmentwicklung die Daten der Simulationen mit und ohne Verwendung der implementierten Nachbarschaftslisten zu vergleichen, um in kurzer Zeit mögliche Fehlerquellen ausfindig machen zu können.

Kapitel 5

Laufzeitoptimierung mithilfe von Nachbarschaftslisten

Die vorliegende Arbeit konzentriert sich zur Laufzeitoptimierung auf die Verwendung sogenannter Nachbarschaftslisten. Diese werden in der Molekulardynamik erfolgreich eingesetzt, um Rechenzeiten für Vielteilchensysteme zu optimieren. Analog zur Molekulardynamik beinhalten die Systeme in der Fußgängerdynamik viele Personen. Die Systeme differenzieren sich jedoch in einigen Aspekten, auf die im Laufe des folgenden Unterkapitels eingegangen wird.

Bei dem vorliegenden, kräftebasierten Modell werden für jede Person treibende und abstoßende Kräfte berechnet, insbesondere die der Personen untereinander. Die auf eine Person wirkende interaktive Kraft ergibt sich dabei aus der Summe der Wechselwirkungen mit allen anderen Personen. Insgesamt sind pro Zeitschritt $N(N - 1)$ Kraftberechnungen nötig, was einer Komplexität von $\mathcal{O}(N^2)$ entspricht, wobei N die Anzahl der Personen ist.

5.1 Grundlagen

Die interaktiven Kräfte verkleinern sich mit größerer Distanz der Fußgänger und sind ab einem bestimmten effektiven Abstand E_{max} vernachlässigbar gering. Dadurch kann um jede Person ein entsprechend großer, kreisförmiger Bereich definiert werden, in dem sich die für die Kraftberechnung dieser Person relevanten Fußgänger befinden. Der Radius dieses Kreises wird als Cutoff-Radius R_c bezeichnet und bezieht sich auf den Abstand der Personen, d. h. die Distanz zwischen den jeweiligen Mittelpunkten (s. Kapitel 3.3.1). Um R_c zu bestimmen, muss berücksichtigt werden, dass der effektive Abstand in Abhängigkeit vom Platzbedarf der Personen variiert (s. Gleichung (3.15)), sodass für R_c der maximal mögliche Abstand ermittelt werden muss. Da der maximale Interaktions-Abstand

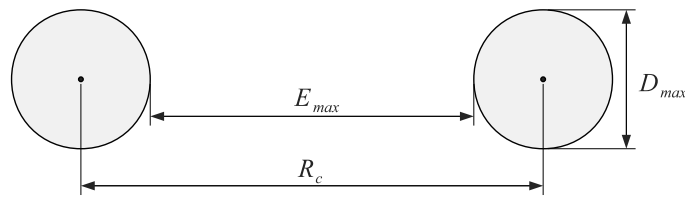


Abbildung 5.1: Bestimmung des Cutoff-Radius unter Berücksichtigung des maximal möglichen Abstands.

vorgegeben ist, setzt sich der Cutoff-Radius aus E_{max} und dem maximalen Platzbedarf zusammen (s. Abbildung 5.1):

$$R_c = E_{max} + D_{max} = E_{max} + D(\vec{v}_{max}). \quad (5.1)$$

Unter Berücksichtigung des Cutoff-Radius müssen maximal $N(N^* - 1)$ Kraftberechnungen durchgeführt werden, wobei $N^* = \rho\pi R_c^2$ gilt und ρ eine angenommene, durchschnittliche Dichte im Simulationsgebiet ist. Im Vergleich zu bestimmten Anwendungen in der Molekulardynamik gilt in der Fußgängerdynamik nicht Newtons drittes Gesetz „*actio=reactio*“, weswegen die Anzahl der Kraftberechnungen nicht halbiert werden kann. Die Komplexität der Kraftberechnungen reduziert sich demzufolge auf $\mathcal{O}(NN^*)$. Da N^* konstant ist und in der Regel $N \gg N^*$, handelt es sich im Wesentlichen um ein Problem der Komplexität $\mathcal{O}(N)$. Um die relevanten Personen innerhalb des Kreises zu ermitteln, ist jedoch weiterhin die Berechnung des Abstandes zu allen anderen Personen nötig. Der Aufwand der Abstandsberechnungen und damit die Gesamt-Komplexität soll von $\mathcal{O}(N^2)$ auf $\mathcal{O}(N)$ verringert werden, indem Nachbarschaftslisten verwendet werden, die die für einen Fußgänger relevanten Personen, die Nachbarn, verwalten.

Es existieren verschiedene Konzepte, die die Eigenschaft von kurzreichweitigen Kräften in Form von Nachbarschaftslisten nutzen. Zwei häufig in der Molekulardynamik verwendete Ansätze sind die Verlet-Listen und die Linked-Cell-Listen, die im Rahmen dieser Arbeit implementiert wurden und hinsichtlich des Optimierungspotentials bzgl. der Laufzeit im Bereich der Fußgängerdynamik in mehreren Testfällen untersucht wurden.

Im Rahmen des Projekts Hermes ist eine Simulationsberechnung auf einem Cell-Prozessor (s. Kapitel 8.2) vorgesehen, bei dem berücksichtigt werden muss, dass der lokale Speicher der einzelnen Prozessoren stark begrenzt ist (256 KB). Daher spielt neben der Minimierung der Laufzeit auch der Speicherbedarf eine wesentliche Rolle.

Das Optimierungspotential von Nachbarschaftslisten in der Fußgängerdynamik zur Effizienzsteigerung von Simulationen wird durch mehrere Gegebenheiten bestimmt. Diese haben abhängig von der jeweiligen Nachbarschaftslisten-Variante sowohl auf die Zeitkomplexität als auch auf den Speicherbedarf des Programms eine unterschiedliche Wirkung:

- **Personenzahl**

Die Zeitkomplexität und der Speicherbedarf wird entscheidend durch die Anzahl der zu simulierenden Fußgänger beeinflusst. Um Prognosen für große Personenzahlen stellen zu können, soll untersucht werden, wie sich die Laufzeit und der Speicherbedarf in Abhängigkeit von der Personenzahl verändert.

- **Homogenität der Verteilung**

In dem Bereich der Fußgängerdynamik handelt es sich um inhomogene Verteilungen, die sich beispielsweise dadurch ergeben, dass Personen in einem Gebäude auf mehrere Räume verteilt sind. Besonders beim Verlassen des Gebäudes entstehen Stausituationen an den Ausgängen und somit Bereiche mit unterschiedlich hohen Dichten. Die Auswirkung solcher Inhomogenitäten auf die Simulation mithilfe von Nachbarschaftslisten muss untersucht werden (s. Kapitel 6.3).

- **Komplexität der Geometrie**

In der Molekulardynamik werden die Nachbarschaftslisten zur Simulation von einfachen, großen Systemen eingesetzt, die durch eine (rechteckige) Simulationsbox mit periodischen Randbedingungen repräsentiert werden. Daher ist zu untersuchen, welchen Einfluss komplexe Gebäudestrukturen auf das Optimierungspotential durch den Einsatz von Nachbarschaftslisten haben.

5.2 Verlet-Listen

Die heute unter dem Namen Verlet-Listen bekannte Technik zur Verwaltung von Nachbarn wurde 1967 von Loup Verlet eingeführt [19] und führt zu einer Gesamt-Komplexität von $\mathcal{O}(N^2)$ mit kleinem Vorfaktor. Für jede Person wird eine Liste angelegt, in der diejenigen Fußgänger gespeichert werden, die sich innerhalb des Cutoff-Radius R_c befinden, sodass zur Kraftberechnung einer Person nur noch die Personen in der Nachbarschaftsliste verwendet werden. Da sich die Positionen der Personen jedoch mit der Zeit ändern, werden Personen sowohl aus der Interaktionsumgebung hinaus- als auch hineinwandern. Eine Aktualisierung der Nachbarschaftslisten in jedem Zeitschritt würde $N(N-1)$ Abstandsberechnungen erfordern. Zur Reduzierung der Anzahl dieser Berechnungen werden in den Listen zusätzlich diejenigen Personen gespeichert, die sich in einem Reservoir befinden, das mit einer Breite R_s , dem Skin-Radius, um den Cutoff-Bereich liegt (s. Abbildung 5.2). Mithilfe dieses Reservoirs reicht es aus, die Listen erst dann zu aktualisieren, wenn eine Person außerhalb des Reservoirs in den Bereich des Cutoff-Radius gelangt. Dadurch beträgt die Komplexität der Abstandsberechnungen zwar immer noch $\mathcal{O}(N^2)$, allerdings mit einem kleinen Vorfaktor. Durch die Einführung des Reservoirs müssen in jedem Zeitschritt nur noch maximal $N(N^+ - 1)$ Abstandsberechnungen durchgeführt werden, mit $N^+ = \rho\pi(R_c + R_s)^2$, um zu überprüfen, ob der effektive Abstand kleiner als E_{max} ist und somit eine Kraftberechnung durchgeführt werden muss. Die Anzahl n der Zeitschritte bis

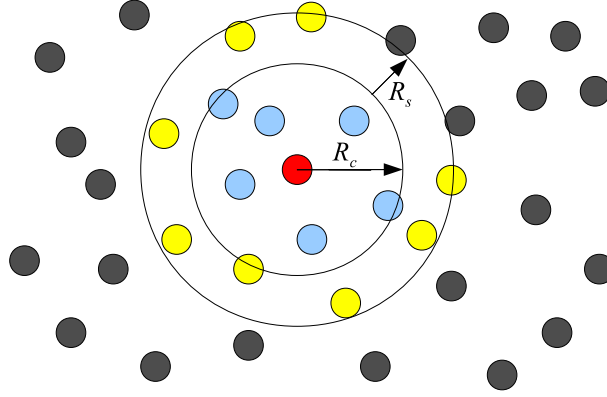


Abbildung 5.2: Qualitative Darstellung des Prinzips der Verlet-Listen: Definition der Nachbarumgebung mit Cutoff-Radius R_c und Skin-Radius R_s . Die blau markierten Personen fließen in die Kraftberechnung ein, die gelben Personen befinden sich im Reservoir und werden ebenfalls in der Nachbarschaftsliste der roten Person gespeichert.

zur nötigen Aktualisierung ergibt sich aus:

$$\max_i \|\vec{x}_i(t_0 + n\Delta t) - \vec{x}_i(t_0)\| > \frac{R_s}{2},$$

wenn $\vec{x}_i(t)$ die Position der i -ten Person zum Zeitpunkt t , t_0 der Zeitpunkt der letzten Aktualisierung und Δt die Größe eines Zeitschrittes ist. Hierbei muss jedoch der variable Platzbedarf der Personen berücksichtigt werden. Im (praktisch unmöglichen) Worst Case gehen zwei ruhende Personen mit maximaler Geschwindigkeit aufeinander zu und beschleunigen dabei sofort von Geschwindigkeit 0 auf v_{max} . Der Zusammenhang ist in Abbildung 5.3 dargestellt. Um diesen Sachverhalt zu verdeutlichen, sind zwei Personen in einem Abstand R_s voneinander positioniert (Abbildung 5.3 links). Eine Aktualisierung ist erforderlich, wenn der effektive Abstand 0 wird. Ohne einen variablen Volumenausschluss muss nur die Geschwindigkeit betrachtet werden (Abbildung 5.3 Mitte). Bei positiver Beschleunigung wächst das Volumen einer Person im Extremfall von D_{min} auf D_{max} , sodass der effektive Abstand bereits zu einem früheren Zeitpunkt 0 wird (Abbildung 5.3 rechts). Daraus folgt die Bedingung für eine Aktualisierung:

$$\max_i \|\vec{x}_i(t_0 + n\Delta t) - \vec{x}_i(t_0)\| > \frac{R_s}{2} - \max_i \left(\frac{D_i - D_{min}}{2} \right).$$

Ein für alle Personen und für die komplette Simulation gültiges n ergibt sich folglich durch

$$n = \left\lceil \frac{R_s - (D_{max} - D_{min})}{2 \Delta t v_{max}} \right\rceil.$$

Der Wert von n und damit auch die Komplexität bezüglich der Abstandsberechnungen wird durch die Größe des Skin-Radius R_s beeinflusst. Mit $R_s = D_{max} - D_{min}$ ist das Reservoir zu klein, sodass in jedem Zeitschritt eine Aktualisierung der Nachbarschaftslisten

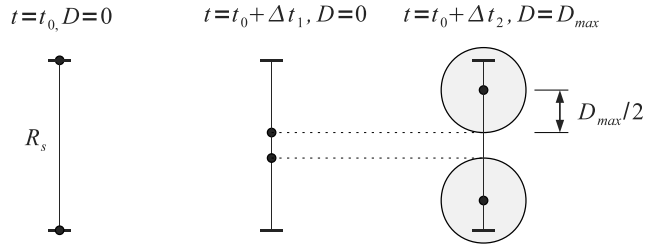


Abbildung 5.3: Skizze zur Darstellung der Abhängigkeit des Aktualisierungsintervalls von der Geschwindigkeit und dem daraus resultierenden Platzbedarf der Fußgänger.

nötig wäre. Bei dem Ermitteln der interaktiven Kräfte würden jedoch unnötige Abstandsberechnungen vermieden. Das andere Extrem ist $R_s = \infty$, sodass nur einmal zu Beginn der Simulation die Listen erstellt werden müssen, aber bei der Kraftberechnung in jedem Zeitschritt der Abstand zu allen anderen Personen berechnet werden muss. In beiden Fällen müssen folglich pro Zeitschritt $N(N - 1)$ Abstände berechnet werden. Nach [18] liegt es nahe, dass ein Wert für R_s existiert, der die Anzahl der Abstandsberechnungen minimiert. Im Kapitel 6.1 wird die Auswirkung der Variation von R_s auf die Rechenzeit untersucht.

Der Speicherbedarf ist bei den Verlet-Listen ein wesentlicher Aspekt, denn für jede Person muss eine Nachbarschaftsliste verwaltet werden, in der alle N^+ Nachbarn gespeichert werden. Dies führt zu einer Speicherkomplexität von $\mathcal{O}(NN^+)$ und zu möglichen Einschränkungen der Personenzahl bei Simulationen auf dem Cell-Prozessor.

Implementierung

Die Implementierung der Listen erfolgt in Form eines zweidimensionalen Arrays, in dem die Indizes der jeweiligen Personen gespeichert werden. Da die Größe von Arrays statisch ist und nur unter unnötigem Aufwand verändert werden kann, wird hier zu Beginn die maximal mögliche Anzahl von Nachbarn ermittelt:

$$N_{max}^+ = \rho_{max} \pi (R_c + R_s)^2.$$

ρ_{max} ist die maximal mögliche Dichte und wurde hier 10 Personen/m² gewählt [13].

Der Ablauf der Nachbarschaftslisten-Aktualisierungen und der Berechnung der abstoßenden interaktiven Kräfte ist schematisch in Form von Nassi-Shneiderman-Diagrammen in Abbildung 5.4 bzw. 5.5 dargestellt. Bei der Implementierung wird statt des Abstands das Quadrat des Abstands berechnet, um die aufwendigere Wurzelberechnung zu vermeiden. Zur Kraftberechnung ist noch hinzuzufügen, dass eine Abstandsabfrage die Dynamik der Fußgänger theoretisch nicht beeinflusst, da die Kraft bei einem effektiven Abstand, der größer als E_{max} ist, hinreichend klein bzw. Null ist. Laufzeitmessungen ergaben jedoch, dass die zusätzliche Abfrage Rechenzeit spart, denn unnötige, rechenintensive Kraftbe-

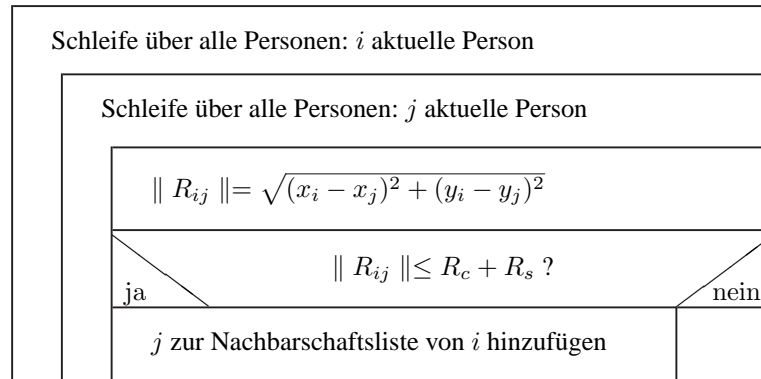


Abbildung 5.4: Aktualisierung der Nachbarschaftslisten bei Verlet.

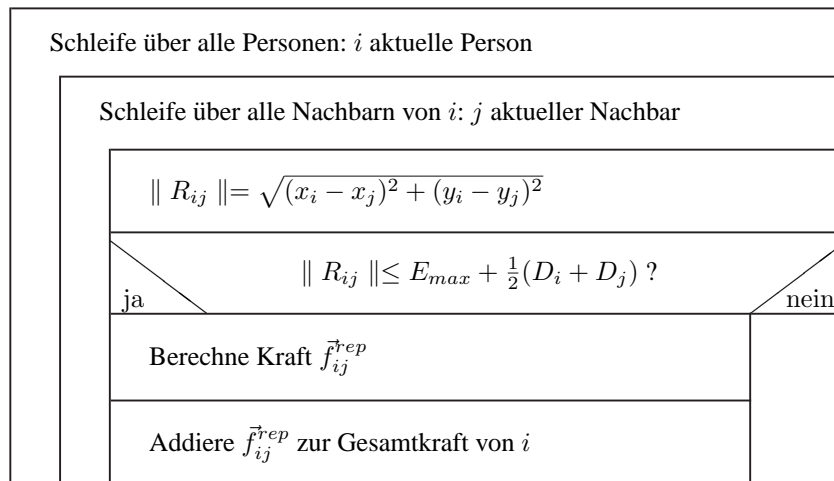


Abbildung 5.5: Berechnungsschritt der abstoßenden Kraft zwischen Personen bei der Verwendung von Verlet-Listen.

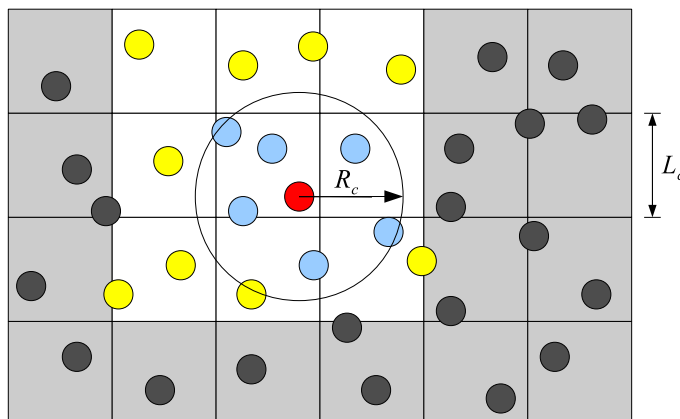


Abbildung 5.6: Qualitative Darstellung des Prinzips der Linked-Cell-Listen: Aufteilung des Simulationsgebiets in Zellen mit Kantenlänge $L_c = R_c$. Die blau markierten Personen fließen in die Kraftberechnung ein, die gelben Personen gehören zur Nachbarschaft der roten Person.

rechnungen werden dadurch vermieden. Im Falle von Personen, die sich näher als E_{max} befinden, wird der zuvor berechnete Abstand für die jeweilige Kraftberechnung erneut verwendet, indem er an die entsprechende Routine übergeben wird.

5.3 Linked-Cell-Listen

Mithilfe von Linked-Cell-Listen [9] wird die Komplexität sowohl der Kraft- als auch der Abstandsberechnungen im Gegensatz zu den Verlet-Listen auf $\mathcal{O}(N)$ reduziert. Bei dieser Methode wird das Simulationsgebiet (in der Molekulardynamik die Simulationsbox) in quadratische Zellen unterteilt, deren Kantenlänge L_c üblicherweise dem Cutoff-Radius R_c entspricht, denn dadurch müssen bei der Kraftberechnung für eine Person nur die Personen in der eigenen und den 8 Nachbarzellen berücksichtigt werden. In Abbildung 5.6 ist das relevante Gebiet exemplarisch für eine Person markiert. Hierdurch wird ähnlich zu den Verlet-Listen der Aufwand für die Kraft- und Abstandsberechnungen reduziert: pro Zeitschritt sind maximal $N(N^+ - 1)$ Abstandsberechnungen ($N^+ = \rho 9 R_c^2$) und maximal $N(N^* - 1)$ Kraftberechnungen ($N^* = \rho \pi R_c^2$) nötig.

Zum Verwalten der Personen in den zugehörigen Zellen werden zwei Listen verwendet. Das Prinzip ist in Abbildung 5.7 dargestellt. Die eine Liste (Head) wird verwendet, um die Zellen zu adressieren und im Falle von dort vorhandenen Personen auf die erste Person zu verweisen. Mithilfe der zweiten Liste zeigt eine Person jeweils auf eine weitere in der Zelle befindliche Person oder markiert das Ende der Verkettung. In Abbildung 5.7 sind exemplarisch ausschnittsweise drei Zellen (Index 14, 15 und 16) mit unterschiedlich vielen Personen (5.7(a)) und der entsprechende Teil in der Liste (5.7(b)) dargestellt und die Verlinkungen visualisiert.

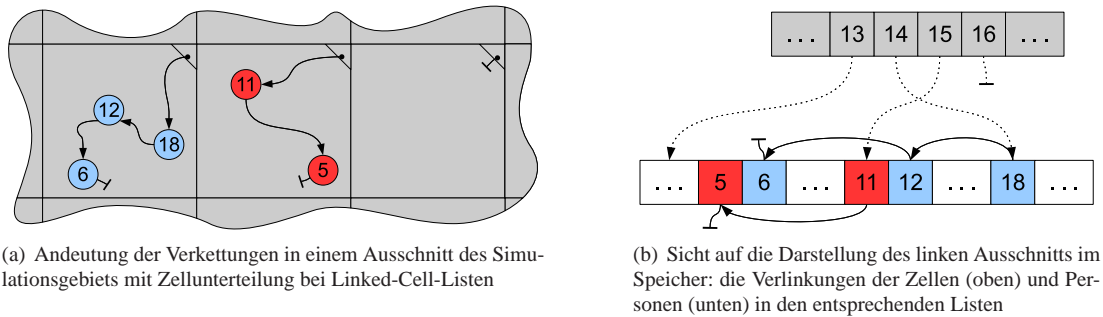


Abbildung 5.7: Schematische Darstellung der Listen, die bei der Methode der Linked-Cell-Listen zur Verwaltung der Personen in den Zellen verwendet werden. In (a) ist ein Ausschnitt des Simulationsgebiets mit unterschiedlicher Besetzung der Zellen zu sehen. Zusätzlich werden die Verlinkungen angedeutet. Die zugehörige Repräsentation im Speicher ist in (b) visualisiert. Dabei zeigen die Einträge der oberen Head-Liste (die Zellen) auf eine Person der jeweiligen Zelle in der zweiten Listen, die wiederum auf eine weitere Person zeigt oder das Ende der Verkettung markiert.

Die Aktualisierung der beiden Listen muss in jedem Zeitschritt erfolgen, da sich jedes Mal eine neue Belegung ergeben kann. Die Komplexität liegt im Gegensatz zu den Verlet-Listen jedoch bei $\mathcal{O}(N)$, da nur einmal über alle Personen iteriert werden muss. Zudem sind hierbei keine Abstandsberechnungen nötig, es muss lediglich für jede Person die zugehörige Zelle bestimmt werden. Abbildung 5.8 zeigt die notwendigen Schritte.

Für die Kraftberechnung wird über alle Zellen iteriert (über die Head-Liste) und jeweils überprüft, ob in der Zelle Personen vorhanden sind. Im positiven Fall wird die Kraft für die jeweilige Person berechnet, indem alle Personen in der aktuellen und den Nachbarzellen betrachtet werden. In der zweiten Liste ist im Eintrag der aktuellen Person die nächste Person der aktuellen Zelle gespeichert. Ist keine weitere Person vorhanden, wird die nächste Zelle betrachtet. Der Algorithmus ist in Abbildung 5.9 dargestellt.

Für die zwei verwendeten Listen benötigt die Head-Liste Speicher für alle Zellen, die zweite Liste für alle Personen. Insgesamt skaliert der Speicherbedarf mit Ordnung $\mathcal{O}(N)$. Der Nachteil dieser Art der Speicherung ist jedoch die indirekte Adressierung der jeweils nächsten Person, sodass Cache-Misses und dadurch ein Performance-Verlust entstehen können. Auf Methoden zur Vermeidung dieses Problems wird in Kapitel 8 eingegangen.

Fußgängerbewegungen werden meist für komplexe Geometriestrukturen simuliert, beispielsweise mehrere Räume. Es liegt daher nahe, pro Raum eine eigene Zellaufteilung und -verwaltung zu implementieren. Besonders für eine spätere Parallelisierung zur weiteren Laufzeitoptimierung wäre ein solcher Ansatz vorteilhaft. Ein weiterer Vorteil wäre, dass Krafteinwirkungen durch Wände hindurch von Personen, die sich zwar innerhalb des Cutoff-Bereichs, aber in einem anderen Raum befinden, automatisch durch die Unabhängigkeit der Zellen der verschiedenen Räume vermieden werden. Allerdings ist zum einen der Overhead zur Verwaltung der Listen pro Raum verhältnismäßig groß und die

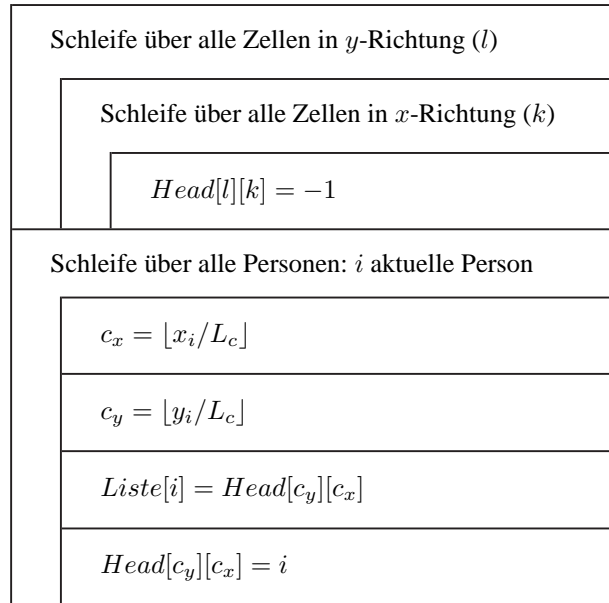


Abbildung 5.8: Aktualisierung der Linked-Cell-Listen.

Effizienz gering, insbesondere, wenn die Räume so klein sind, dass sie nur wenige Zellen beinhalten. Zum anderen stellen die Übergänge zwischen den Räumen ein Problem dar. Ähnlich wie bei einer Gebietszerlegung im Bereich der Parallelisierung müssten auch hier sogenannte Geister-Zellen¹ mitgeführt werden, um bei den Übergängen (d. h. Türen bzw. Ausgängen) die Krafteinwirkungen der Personen des jeweils anderen Raumes berücksichtigen zu können. Hinzu kommt, dass wegen eventueller lokaler Koordinatensysteme pro Raum und wegen versetzter Zellen (bedingt durch die unterschiedliche Größe der Räume) viele Sonderfälle berücksichtigt werden müssten. Aufgrund der genannten Probleme wird in dieser Arbeit für das gesamte System ein globales Koordinatensystem verwendet und ein rechteckiges und somit einfach zu verwaltendes Zellgitter mit $L_c = R_c$ über die komplette Raumgeometrie gelegt, wie in Abbildung 5.10 exemplarisch dargestellt ist. Für zukünftige komplexere Szenarien, bei denen das System sich über mehrere Ebenen erstreckt („dreidimensional“), ist jedoch eine Zellverwaltung pro Ebene ratsam.

Implementierung

In der Molekulardynamik wird häufig eine Simulationsbox mit periodischen Randbedingungen definiert [1]. In der Fußgängerdynamik werden meist Systeme mit geschlossenen oder offenen Randbedingungen betrachtet. Um eine notwendige Überprüfung zur Bestim-

¹In der Parallelisierung wird bei einer Gebietszerlegung jedem Prozess ein Teilgebiet zugeordnet. Häufig bestehen bei den Berechnungen lokale Abhängigkeiten, daher benötigen benachbarte Prozesse die Daten an den Rändern des Teilgebiets des jeweils angrenzenden Prozesses. Aufgrund der räumlichen Diskretisierung werden diese überlappenden Randbereiche Geister-Zellen genannt, da sie nicht zu dem eigentlichen Teilbereich des Prozesses gehören.

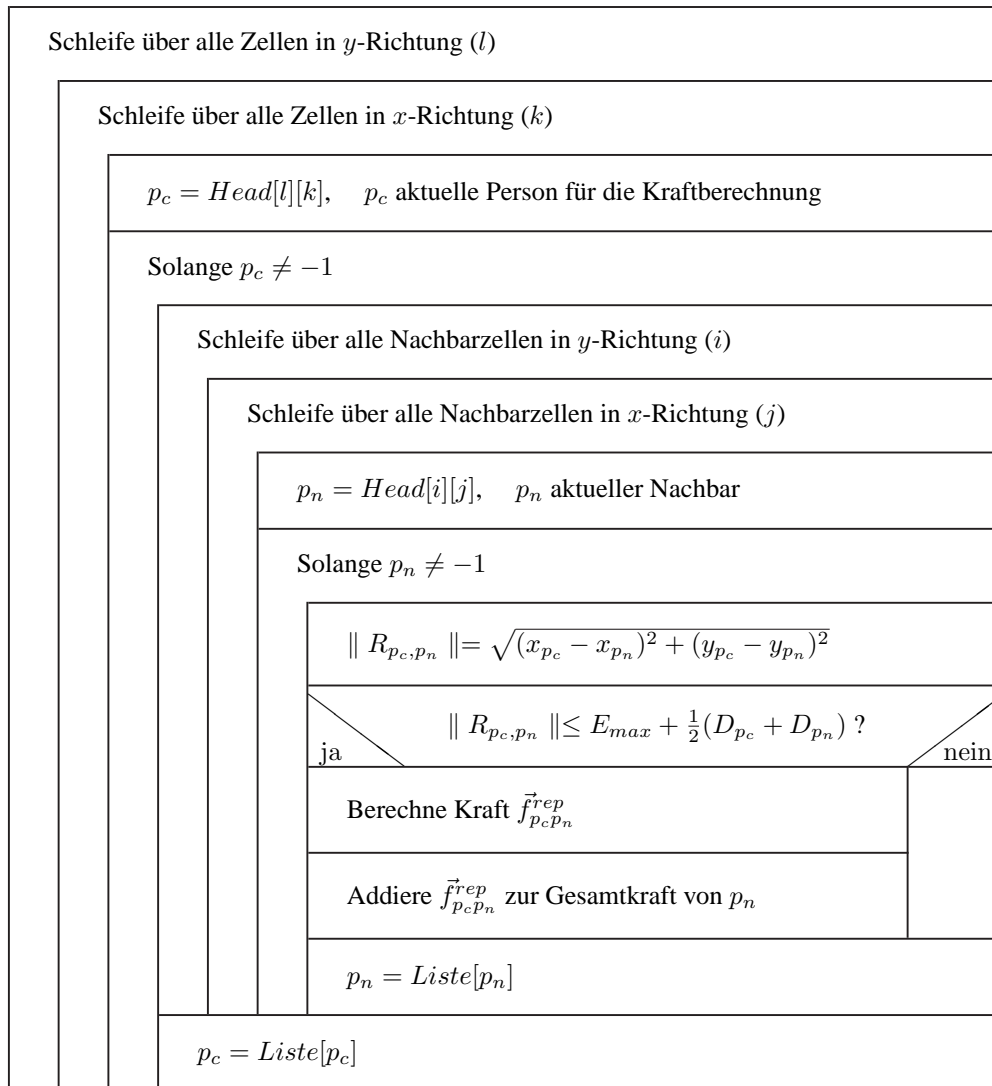


Abbildung 5.9: Berechnung der abstoßenden Kraft zwischen Personen bei der Verwendung von Linked-Cell-Listen.

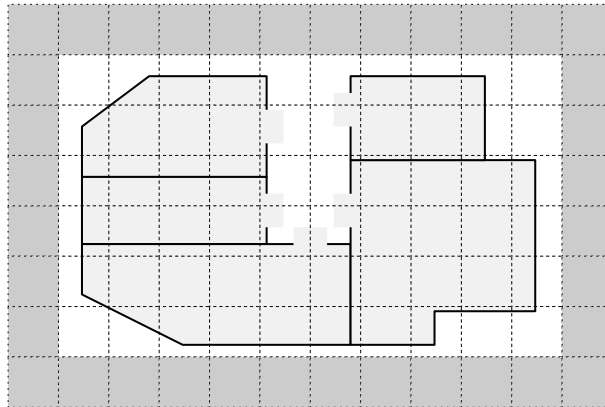


Abbildung 5.10: Exemplarische Darstellung einer Zellaufteilung für Linked-Cell-Listen durch Überlagerung eines Gitters über die gesamte Geometrie. Zusätzlich werden an den Rändern „Pseudo-Zellen“ hinzugefügt (in grau unterlegt).

mung der jeweiligen Nachbarzellen im Falle der Randzellen zu vermeiden, wurden hier „Pseudo-Randzellen“ eingeführt, also das Gitter an jedem Rand um eine Zelle erweitert (s. Abbildung 5.10). Der erforderliche Speicherbedarf ist gering, aber die resultierende Performance ist größer.

Die Head-Liste ist in Form eines zweidimensionalen Arrays implementiert, die zweite Liste ist ein eindimensionales Array. Beide speichern die Indizes der Personen bzw. -1 , wenn die Zelle leer ist oder wenn es sich um die Person am Ende der Verkettung handelt.

Wie bei den Verlet-Listen erfolgt auch bei den Linked-Cell-Listen in der Routine zur Kraftberechnung eine Abfrage, ob der effektive Abstand zum Nachbarn kleiner als E_{max} ist, um Rechenzeit im Vergleich zu sonst unnötigen Kraftberechnungen zu sparen (s. Kapitel 5.2).

Kapitel 6

Performancetests

Mithilfe von Performancetests wurde untersucht, wie stark die Laufzeit einer Simulation durch den Einsatz von Nachbarschaftslisten verkürzt werden kann. Dazu wurde neben der Laufzeit der gewinnbringende Faktor bezüglich der Brute-Force-Methode betrachtet. Dieser Faktor wird als Speedup S bezeichnet und ist definiert durch

$$S = \frac{t_{BF}}{t_{NL}}, \quad (6.1)$$

wobei t die Laufzeit bei Verwendung der Brute-Force-Methode (BF) und der jeweiligen Nachbarschaftsliste (NL) ist.

Der Speedup hängt von mehreren Eigenschaften der Simulation ab, etwa der Anzahl, der Dichte oder der Verteilung der simulierten Fußgänger im Raum. Verschiedene Testszenarien wurden konstruiert, um den Einfluss der unterschiedlichen Gegebenheiten auf den Speedup zu untersuchen. Dabei interessiert insbesondere die Skalierung im Hinblick auf große Personenzahlen.

Um einen Vergleich zwischen verschiedenen Tests zu ermöglichen, wurde in allen Fällen dieselbe simulierte Echtzeit gewählt. Für ausreichend viele Berechnungen wurden 10 000 Iterationsschritte festgelegt, sodass aus einem Zeitintervall $\Delta t = 0,01$ s eine simulierte Echtzeit von $t_{sim} = 100$ s resultiert. Des Weiteren wurde pro Simulationskonfiguration dieselbe Anfangsverteilung der Personen gewählt. Die Laufzeitmessungen wurden unter dem Betriebssystem openSUSE 10.2 auf einem Intel Pentium D Prozessor mit 3,40 GHz, einem L2-Cache der Größe 2048 KB und einem 2048 MB großen Hauptspeicher durchgeführt.

Das Optimierungspotential durch die Verwendung von Nachbarschaftslisten wurde bei Variation verschiedener Parameter in zwei Hauptszenarien untersucht: einer homogenen und einer inhomogenen Verteilung der Fußgänger. Hierbei ist insbesondere das Laufzeitverhalten bei inhomogenen räumlichen Personenverteilungen interessant, da Bereiche unterschiedlich hoher Dichte in der Fußgängerdynamik häufig auftreten.

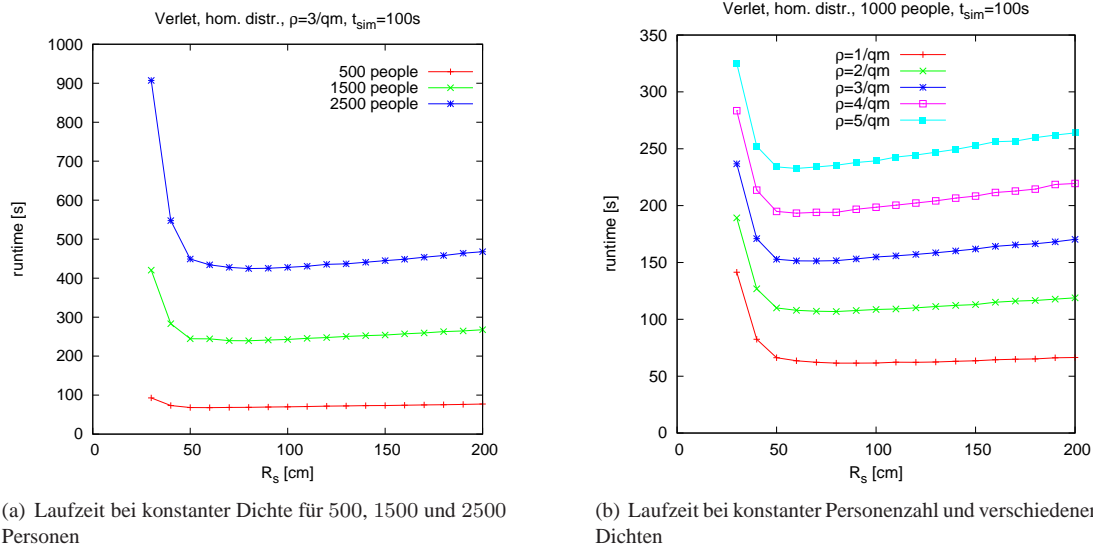


Abbildung 6.1: Laufzeitmessungen bei Verwendung der Verlet-Listen mit Variation des Skin-Radius R_s in einer homogenen Verteilung.

Zuvor wurde für die Verlet-Listen ein möglichst optimaler Wert für den noch zu bestimmenden Skin-Radius R_s ermittelt, der für die darauffolgenden Tests verwendet wurde.

6.1 Bestimmung des Skin-Radius

In Kapitel 5.2 wurde vermutet, dass bei den Verlet-Listen für den Skin-Radius ein optimaler Wert existiert. Dieser hängt sowohl von der Anzahl der simulierten Personen als auch von der Dichte ab. Daher wurde die Laufzeit bei Variation des Skin-Radius in 10 cm-Intervallen zum einen bei konstanter Dichte für verschiedene Personenzahlen (s. Abbildung 6.1(a)) und zum anderen bei fester Personenzahl für unterschiedliche Dichten gemessen (s. Abbildung 6.1(b)).

In Abbildung 6.1(a) ist zu sehen, dass bis $R_s = 50$ cm ein besonders bei größeren Personenzahlen deutlicher Abfall der Laufzeit zu verzeichnen ist, wobei diese für größere Skin-Radien langsam steigt. Weiterhin ist zu erkennen, dass sich der optimale Skin-Radius für weniger Personen verringert, was andere, hier nicht abgebildete Messungen für 50, 150 und 250 Personen bestätigen. Das Minimum für 2500 Personen liegt bei einem Skin-Radius von ca. 80 cm.

Bei der Untersuchung des optimalen Skin-Radius für verschiedene Dichten wird in Abbildung 6.1(b) deutlich, dass sich dieser mit steigender Dichte verkleinert. Ebenfalls ist zu bemerken, dass die Laufzeit bei höheren Dichten nach dem Minimum für größere Skin-Radien stärker ansteigt als bei niedrigen Dichten. So wird beispielsweise bei einer Dichte

von 5 Personen/m² für $R_s = 160$ cm eine ähnliche Laufzeit wie für $R_s = 40$ cm erreicht. Bei Dichten ab 2 Personen/m² liegt der optimale Skin-Radius etwa in einem Bereich zwischen 60 cm und 80 cm.

Da bei Evakuierungen vermehrt höhere Dichten zu erwarten sind, jedoch auch viele Personen simuliert werden sollen, wurden für die nachfolgenden Tests die Verlet-Listen mit einem Skin-Radius von 70 cm verwendet. Im konkreten Anwendungsfall sollte für ein optimales Laufzeitverhalten das jeweils vorliegende Szenario im Hinblick auf die Personenzahl und die zu erwartenden Dichten untersucht werden.

6.2 Homogene Verteilung

Eine homogene Verteilung ist gegeben, wenn die lokale Dichte auf dem gesamten Gebiet, hier einem rechteckigen Raum, konstant ist. Dies wird zu Beginn der Simulation approximativ durch die zufällige Verteilung der Fußgänger erreicht. Da jedoch die Berechnung der wechselwirkenden Kräfte der Hauptbestandteil der Rechenzeit ist, müssen sich die Fußgänger bewegen, um eine Kraft ungleich Null zu erzeugen (s. Gleichung (3.5)). Für eine Bewegung der Fußgänger ist eine treibende Kraft erforderlich, die die Angabe einer Zielrichtung benötigt. Um eine Homogenität während der gesamten Simulation zu gewährleisten, wird in jedem Iterationsschritt pro Fußgänger eine von acht (zwei horizontalen, zwei vertikalen und vier diagonalen) Richtungen zufällig gewählt.

Bei diesem Testszenario ist das Optimierungspotential durch die Verwendung von Nachbarschaftslisten von drei Parametern abhängig: der Dichte ρ , der Größe des Raums A_{room} und der Anzahl der Personen N . Dabei besteht der folgende Zusammenhang:

$$\rho = \frac{N}{A_{room}}.$$

Das heißt, einer der drei Parameter wird durch die Angabe der zwei anderen bestimmt. Bei der Berechnung der Raumgröße aus Dichte und Personenzahl wird ein quadratischer Raum angenommen.

Abgesehen von der Laufzeit der optimierten Simulation ist der in Gleichung (6.1) definierte Speedup interessant. Insbesondere stellt sich die Frage, ob und wie der Speedup in Korrelation zu den variierten Parametern steht. Da der wesentliche Teil der Rechenzeit die Berechnung der interaktiven Kräfte zwischen Fußgängern ist, wird für eine grobe Abschätzung der Speedup durch die Anzahl der entsprechend durchgeführten Kraftbe-

rechnungen ausgedrückt:

$$\begin{aligned}
 S &= \frac{t_{BF}}{t_{NL}} \\
 &\approx \frac{N(N-1)}{N(N^*-1)} \\
 &= \frac{N-1}{\rho \pi R_c^2 - 1} \tag{6.2}
 \end{aligned}$$

$$= \frac{N-1}{\frac{N}{A_{room}} \pi R_c^2 - 1} \xrightarrow{N \rightarrow \infty} \frac{A_{room}}{\pi R_c^2} \tag{6.3}$$

Hieraus lässt sich folgern, dass

1. bei konstanter Dichte und $N \rightarrow \infty$ gilt: $S \rightarrow \infty$ (Gl. (6.2))
2. bei konstanter Personenzahl und $\rho \rightarrow \infty$ gilt: $S \rightarrow 0$ (Gl. (6.2))
3. bei fester Raumgröße und $N \rightarrow \infty$ gilt: $S \rightarrow const.$ (Gl. (6.3))

6.2.1 Testszenario 1: N variiert, ρ konstant

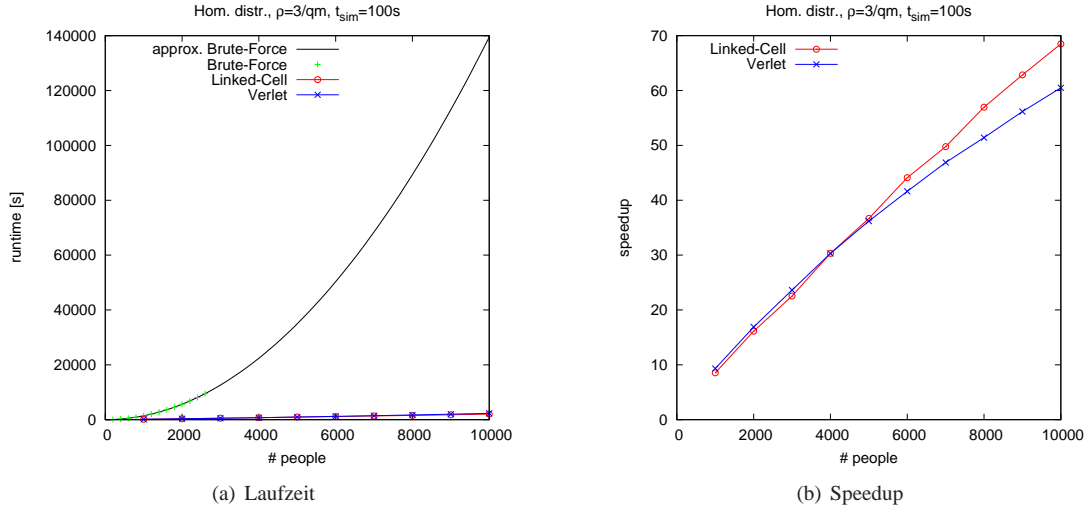


Abbildung 6.2: Laufzeit und Speedup bei homogener Verteilung und konstanter Dichte in Abhängigkeit von der Personenzahl. Dabei werden die Werte der Brute-Force-Methode durch ein Ausgleichspolynom 2. Grades approximiert.

Für diesen Test wurde eine Dichte von 3 Personen/ m^2 gewählt und die Personenzahl von 1000 bis 10 000 in Schritten von 1000 variiert. Für die Brute-Force-Methode wurde aufgrund hoher Laufzeiten in 200er Schritten von 200 bis 2600 simuliert und durch die

Messergebnisse mittels der Methode der kleinsten Fehlerquadrate ein Polynom 2. Grades gelegt.

Das Laufzeitverhalten bei konstanter Dichte und steigender Personenzahl ist offensichtlich. Wie in Abbildung 6.2(a) zu sehen, skaliert aufgrund der $\mathcal{O}(N^2)$ -Komplexität die Laufzeit für die Brute-Force-Methode quadratisch. Aus der konstanten Dichte resultiert eine feste Anzahl der für die wechselwirkende Kraft relevanten Nachbarn, sodass bei den Nachbarschaftslisten der Aufwand für die Kraftberechnungen linear steigt. Allerdings wird beim Vergleich der Unterschied zwischen den beiden Nachbarschaftslisten für große N deutlich, da die Gesamtkomplexität der Verlet-Listen aufgrund der Abstandsrechnungen $\mathcal{O}(N^2)$, bei den Linked-Cell-Listen jedoch $\mathcal{O}(N)$ ist. Dies ist in Abbildung 6.2(b) zu erkennen, in der der Speedup der beiden Verfahren dargestellt ist. Dieser steigt für die Linked-Cell-Listen linear und erreicht für 10 000 Personen ca. den Wert 70. Dahingegen wächst der Speedup für die Verlet-Listen wegen der quadratischen Skalierung der Laufzeit schwächer als linear und ist bei 10 000 Personen ca. 60 mal schneller als die Brute-Force-Methode. Darüber hinaus fällt auf, dass durch beide Nachbarschaftslisten für N bis ca. 5000 eine ähnliche Performance resultiert. Die Linked-Cell-Listen erreichen dahingegen wegen der linearen Skalierung ab ca. 5000 Personen einen deutlich größeren Vorsprung, der mit wachsendem N weiter ausgebaut wird. Es zeigt sich zudem, dass die Messungen mit den Erwartungen übereinstimmen, dass für $\rho = \text{const.}$ und $N \rightarrow \infty$ gilt: $S \rightarrow \infty$.

6.2.2 Testszenario 2: N konstant, ρ variiert

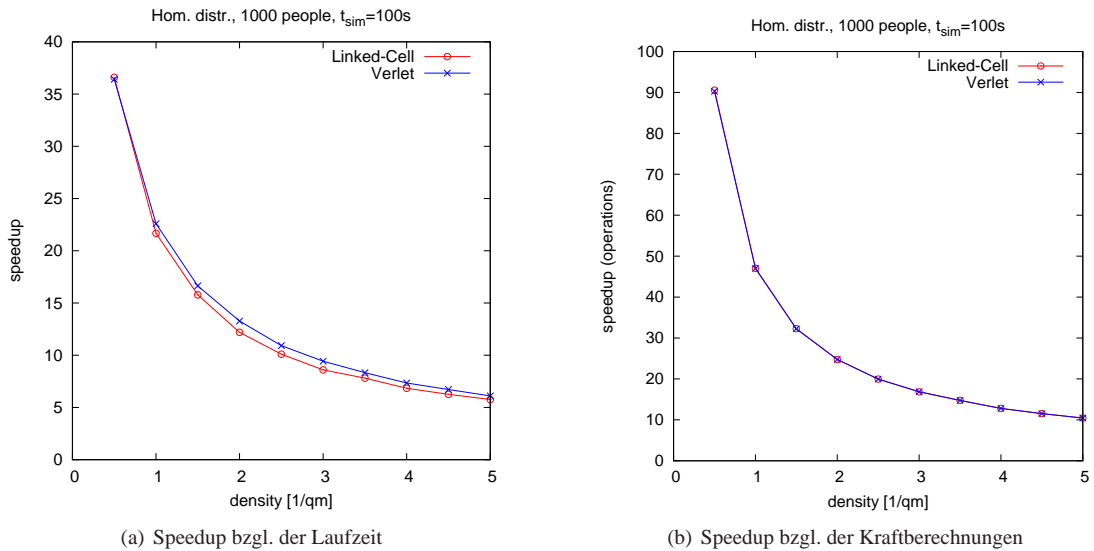


Abbildung 6.3: Speedup der Laufzeit und der Kraftberechnungen bei homogener Verteilung und konstanter Personenzahl in Abhängigkeit von der Dichte.

In diesem Testfall wird $N = 1000$ gewählt und die Dichte von 0,5 bis 5 Personen/m² in 0,5er Schritten variiert. Aufgrund der obigen Abschätzung des Speedups ist zu erwarten, dass dieser mit steigender Dichte abfällt. Da unendlich große Dichten aufgrund des Volumenausschlusses nicht möglich sind, wird der Speedup in der Realität nicht zu Null konvergieren.

In Abbildung 6.3(a) wird deutlich, dass sich bei den Messungen das erwartete Verhalten des Speedups ergibt. Des Weiteren ist zu bemerken, dass der Speedup der Linked-Cell-Listen etwas geringer ist als der der Verlet-Listen ist. Um zu veranschaulichen, dass die Laufzeit maßgeblich von der Anzahl der wechselwirkenden Kraftberechnungen abhängt, ist in Abbildung 6.3(b) der entsprechende Speedup dieser Operationen dargestellt. Dieser zeigt dasselbe Verhalten wie der Speedup bezüglich der Laufzeit. Da die Anzahl der Kraftberechnungen bei beiden Nachbarschaftslisten identisch ist, ergibt sich für beide Varianten derselbe Speedup. Unterschiede in der Laufzeit ergeben sich durch die Anzahl der Abstandsberechnungen und den Aufwand zur Verwaltung der Listen.

Die Laufzeit der Brute-Force-Methode ist unabhängig von der Dichte, da hierbei die Anzahl der Kraftberechnungen konstant ist, und beträgt für 1000 Personen ca. 1400 Sekunden. Mit höherer Dichte steigt sie minimal an, was dadurch zu erklären ist, dass vermehrt Interpolationen zur Kraftberechnung durchgeführt werden müssen (s. Kapitel 3.3.1). Die von den Nachbarschaftslisten benötigte Rechenzeit steigt von 39 Sekunden (Verlet) bzw. 38 Sekunden (Linked-Cell) bei 0,5 Personen/m² auf 236 bzw. 249 Sekunden bei 5 Personen/m².

6.2.3 Testszenario 3: N variiert, A_{room} konstant

Der vielleicht interessanteste Testfall ist die Variation der Personenzahl bei einer konstanten Raumgröße, denn im konkreten Anwendungsfall zur Simulation von Evakuierungen in der *ESPRIT arena* ist die Geometrie fest vorgegeben. Nach obiger Abschätzung wird erwartet, dass der Speedup zu einem konstanten Wert konvergiert. Wie im ersten Testfall werden die Anzahl der Personen von 1000 bis 10 000 variiert und die Ergebnisse der Brute-Force-Methode aufgrund hoher Laufzeiten mittels einer Ausgleichsrechnung approximiert. Hierfür wurden in 200er Schritten Simulationen bis $N = 2600$ durchgeführt. Im Hinblick auf einen späteren Vergleich zum inhomogenen Szenario wird die Raumgröße auf 35 m \times 50 m festgelegt.

Bei der Darstellung des Speedups bzgl. der Kraftberechnungen in Abbildung 6.4(b) wird deutlich, dass bereits für kleine N ein konstanter Wert angenommen wird. Dieser kann mithilfe obiger Abschätzung ermittelt werden. Hierfür ist zu beachten, dass für die Berechnung von R_c der Worst Case angenommen wurde (s. Gleichung (5.1)). Mit $\|\vec{v}_{max}\| =$

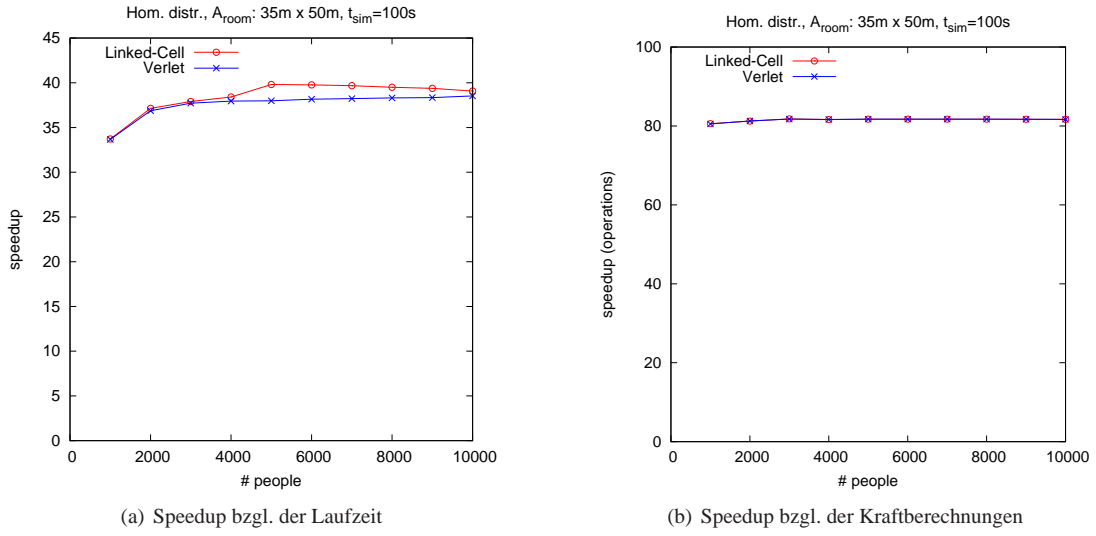


Abbildung 6.4: Speedup der Laufzeit und der Kraftberechnungen bei homogener Verteilung und fester Raumgröße in Abhängigkeit von der Personenzahl.

1,6 m/s ergibt sich ein Wert von

$$\begin{aligned}
 \frac{A_{room}}{\pi R_c^2} &= \frac{A_{room}}{\pi (E_{max} + D(\vec{v}_{max}))^2} \\
 &= \frac{35m \cdot 50m}{\pi (2,5m + (0,2m + 0,2 \cdot 1,6m))^2} \\
 &= \frac{35m \cdot 50m}{\pi (2,5m + 0,52m)^2} \\
 &\approx 61.
 \end{aligned}$$

Der gemessene Speedup liegt jedoch bei ca. 80. Dies ist mit der geringen Geschwindigkeit der Personen in diesem Testszenario zu begründen. Im Best Case ist daher die Geschwindigkeit nahezu Null und daher $D \approx D_{min} = 0,2m$ und

$$\frac{A_{room}}{\pi R_c^2} \approx 76,4,$$

was im Bereich der gemessenen Ergebnisse liegt. Der letztendlich höhere Wert resultiert vermutlich aus der geringeren Anzahl von Kraftberechnungen bei Personen, die sich in dem Randbereich des simulierten Raumes befinden.

Bei der Betrachtung des Speedups bzgl. der Laufzeit in Abbildung 6.4(a) fällt auf, dass dieser erst bei ca. 2000-3000 Personen konvergiert. Bei geringer Personenzahl führen vermutlich der bis dorthin überwiegende Aufwand zur Verwaltung der Nachbarschaftslisten und die indirekte Adressierung zu diesem Verhalten. Zusätzlich zeigt sich, dass die

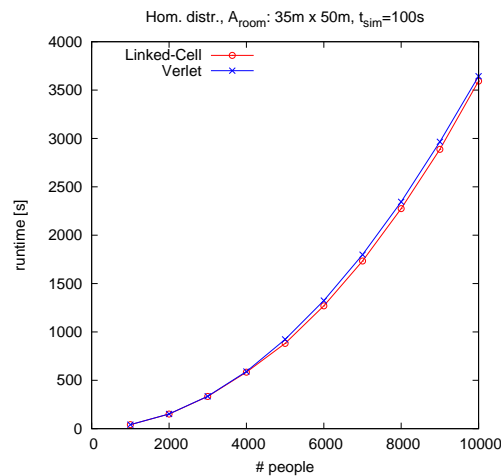


Abbildung 6.5: Laufzeit der Simulationen mit Linked-Cell-Listen und Verlet-Listen bei homogener Verteilung und fester Raumgröße in Abhängigkeit von der Personenzahl.

Linked-Cell-Listen die Laufzeit in einem höheren Maße optimieren als die Verlet-Listen. Der Anstieg des Speedups bei den Linked-Cell-Listen bei $N = 5000$, sowie der ab dort abfallende Speedup kann zu diesem Zeitpunkt nicht begründet werden und bedarf weiterer Untersuchungen.

In Abbildung 6.5 ist die Laufzeit der Simulationen bei Verwendung der beiden implementierten Nachbarschaftslisten dargestellt. Das superlineare Wachstum resultiert aus der linearen Skalierung durch eine steigende Dichte und dem Laufzeitzuwachs, der durch größere Personenzahlen entsteht (s. vorherige Testfälle).

6.3 Inhomogene Verteilung

Eine Betrachtung des Laufzeitgewinns durch Nachbarschaftslisten ist besonders für inhomogene Verteilungen der Fußgänger interessant, da bei Evakuierungen aufgrund von Engstellen lokale Stausituationen mit hoher Dichte entstehen.

Für die Simulationen wird ein Testszenario mit mehreren Engstellen, also Ausgängen, konstruiert. Die Personenzahl muss während der gesamten Simulation konstant bleiben, um Einflüsse auf die Laufzeit wegen Verlassens des Systems auszuschließen. Ein offenes System, bei dem die Personen das Simulationsgebiet verlassen und dadurch nicht mehr zur Kraftberechnung beitragen, ist ungeeignet.

Um eine konstante Personenzahl bei offenen Randbedingungen zu erfüllen, wird ein periodisches System definiert, das durch einen Raum mit 5 Ausgängen an einer der vier Wände realisiert ist (s. Abbildung 6.6). Die Personen werden zu Beginn in dem Raum zufällig verteilt (s. Abbildung 6.6(a)) und verlassen den Raum durch die Ausgänge. Nach

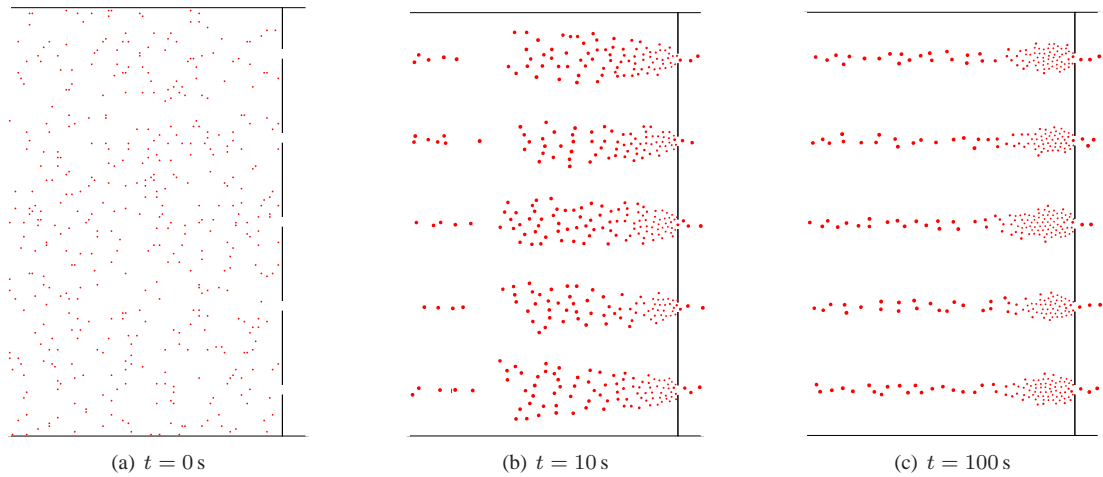


Abbildung 6.6: Szenario für eine inhomogene Verteilung realisiert durch ein periodisches System mit einem Raum und 5 Ausgängen zu drei verschiedenen Zeitpunkten. In (a) ist die Verteilung beim Simulationsstart dargestellt, in (b) das Ergebnis nach 10 s und in (c) nach 100 s. Die Größe der Kreise spiegelt den Platzbedarf der Fußgänger wieder, der mit der Geschwindigkeit skaliert, die zu Beginn Null ist.

dem Durchqueren eines Ausgangs werden sie wieder in den Raum auf die gegenüberliegende Seite positioniert (s. Abbildung 6.6(b)). Um eine korrekte Dynamik an der Engstelle zu gewährleisten, ist dabei wichtig, dass die Fußgänger nach einem Ausgang noch mindestens so weit gehen wie sie Einfluss auf die Personen vor dem Ausgang haben. Das bedeutet, dass diese Mindestdistanz E_{max} entspricht. Hier wurde dieser Abstand auf 3 m gesetzt, um diese Bedingung ausreichend zu erfüllen. Periodisch wirkende Kräfte an den Rändern des Simulationsgebiets wurden in der Implementierung zum Zeitpunkt der Messungen nicht berücksichtigt. Da sich die Personen an den Rändern jedoch mit nahezu Wunschgeschwindigkeit bewegen und die Dichte gering ist, sollten die fehlenden Kraftberechnungen die Dynamik kaum beeinflussen und für die Laufzeitmessungen nicht relevant sein. Allerdings muss darauf geachtet werden, dass der Rückstau durch die Ausgänge keine Wirkung auf die wieder eingesetzten Fußgänger hat. Daher werden in dem Simulationsgebiet der Größe $35 \text{ m} \times 50 \text{ m}$ maximal 2500 Personen positioniert. Die Gesamtfläche entspricht somit der Fläche des Raumes beim homogenen Szenario mit fester Raumgröße.

Für die Laufzeitmessungen wurde die Personenzahl von 100 bis 2500 in 100er Schritten variiert. Des Weiteren wurde eine Ausgangsbreite von 1 m gewählt. Die Ergebnisse werden in Abbildung 6.7(a) gezeigt, in der das $\mathcal{O}(N^2)$ -Verhalten der Brute-Force-Methode deutlich wird. Für $N > 1500$ wird erneut eine Ausgleichsrechnung verwendet. Die Laufzeit der Simulationen mit den Nachbarschaftslisten skaliert hingegen linear. Dies wird besonders in Abbildung 6.7(b) deutlich, in der der Speedup dargestellt wird. Die lineare Skalierung ist dadurch zu begründen, dass wegen der Staubereiche vor den Ausgängen eine Verteilung mit quasi-konstanter Dichte herrscht. In Kapitel 6.2 wurde bereits gezeigt,

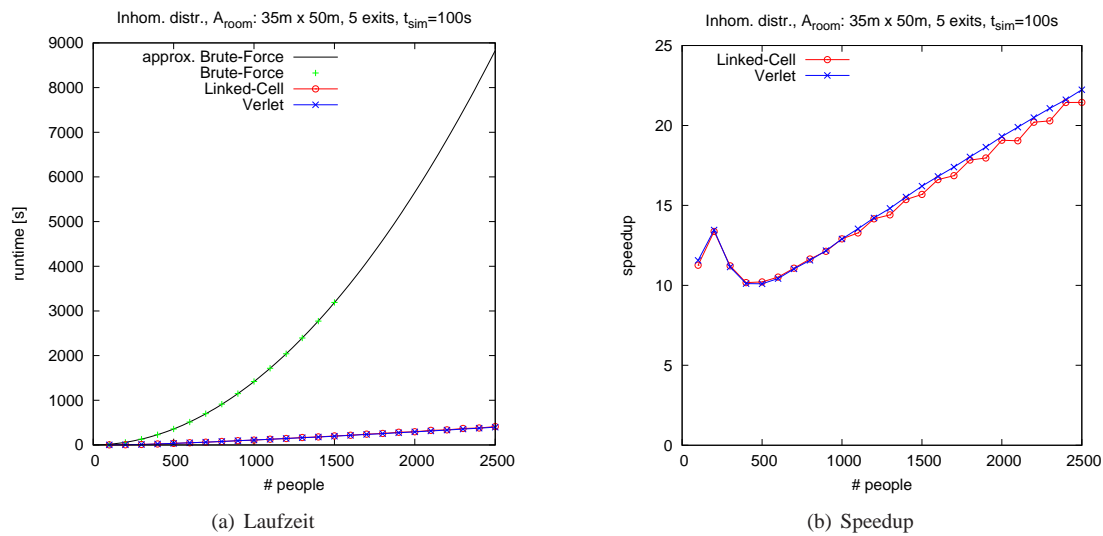


Abbildung 6.7: Laufzeit und Speedup bei inhomogener Verteilung in Abhängigkeit von der Personenzahl. Durch die Staubereiche herrscht eine quasi-konstante Dichte, weswegen der Speedup ähnlich zu dem homogenen Testszenario linear steigt. Bemerkenswert ist das Verhalten des Speedups für Personenzahlen kleiner als 500.

dass bei einer homogenen Verteilung und konstanter Dichte ein mit der Personenzahl stetig steigender Speedup entsteht. Auffallend sind die regelmäßigen Schwankungen des Speedups der Linked-Cell-Listen für $N > 1000$. Eine mögliche Erklärung könnten variierende Speicherzugriffe sein. Weitere Untersuchungen sind nötig, um die genaue Ursache zu ermitteln.

Bemerkenswert ist jedoch der Verlauf des Speedups für weniger als 500 Personen. Für eine detaillierte Analyse wurden die Laufzeiten in diesem Bereich mit Schrittweiten von 25 Personen für jeweils zehn Simulationen gemessen und anschließend gemittelt (s. Abbildung 6.8). Hier ist ein Anstieg des Speedups bis ca. 150 Personen zu verzeichnen, wo es zu einem lokalen Maximum kommt. Nach einem Abfall stellt sich ein lokales Minimum bei ca. 400-450 Personen ein.

Die Begründung des Verlaufs des Speedups liegt in der Staubildung an den Ausgängen, was in Abbildung 6.9 nachvollzogen werden kann. Dort sind die Fußgänger bei einer Momentaufnahme am Ende der jeweiligen Simulation, also nach 100 s Echtzeit, dargestellt und zusätzlich exemplarisch der Cutoff-Bereich eingezeichnet. Bei weniger als 150 Personen entstehen keine bzw. kaum Staus, sodass insbesondere aufgrund der Anordnung der Personen in Form von Linien bei den Nachbarschaftslisten wenige wechselwirkende Kräfte zu berechnen sind (s. Abbildung 6.9(a)). Sobald sich Stauungen bilden, nimmt die Anzahl der interaktiven Kräfte zwischen Fußgängern zu (s. Abbildung 6.9(b)). Ist der Staubereich in der Größenordnung R_c , finden die Kraftberechnungen im Wesentlichen im Staubereich statt und der Performancegewinn durch Verwendung von Nachbarschaftslisten

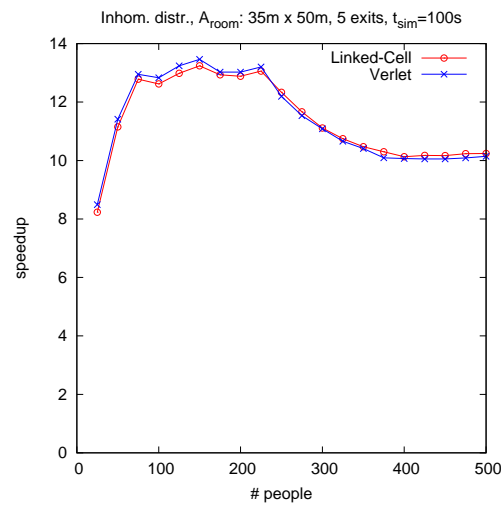


Abbildung 6.8: Speedup bei inhomogener Verteilung in Abhängigkeit von der Personenzahl bis 500 Personen.

ten ist gering. Mit der steigenden Ausdehnung der Staugebiete bei mehr Personen nimmt die Effektivität der Nachbarschaftslisten zu (s. Abbildung 6.9(c)). Ab ca. 500 Personen überwiegt der Anteil der im Stau stehenden Personen im Vergleich zu den sich frei bewegendenden Personen, sodass es sich, wie bereits oben erwähnt, hauptsächlich um eine Verteilung der Fußgänger mit quasi-konstanter Dichte handelt und der Speedup entsprechend skaliert.

Vergleich mit homogener Verteilung

Der Vergleich des Speedups der homogenen und inhomogenen Verteilung (s. Abbildung 6.10) zeigt, dass durch die homogene Verteilung auf dem gesamten Simulationsgebiet eine minimale Anzahl von wechselwirkenden Kräften entsteht. Dies führt zu einem steilen Anstieg des Speedups für $N < 1000$ und im Vergleich zum inhomogenen Testszenario zu einer deutlich höheren Performance. Interessant ist, dass die regelmäßigen Schwankungen beim Linked-Cell-Verfahren sowohl im homogenen als auch im inhomogenen Testszenario auftreten. Dies weist darauf hin, dass nicht die Dynamik der Personenbewegung die Ursache für die regelmäßigen Schwankungen ist.

Da der Speedup bei beiden Nachbarschaftslisten im homogenen Fall konvergiert, wäre eine Weiterentwicklung im inhomogenen Szenario interessant. Aufgrund der festen Raumgröße und des Volumenausschlusses der Personen entsteht aus der inhomogenen Verteilung für steigende Personenzahlen im Grenzfall eine homogene Verteilung. Dadurch ist zu erwarten, dass der Speedup in diesem Fall mit dem des homogenen Szenarios übereinstimmt.

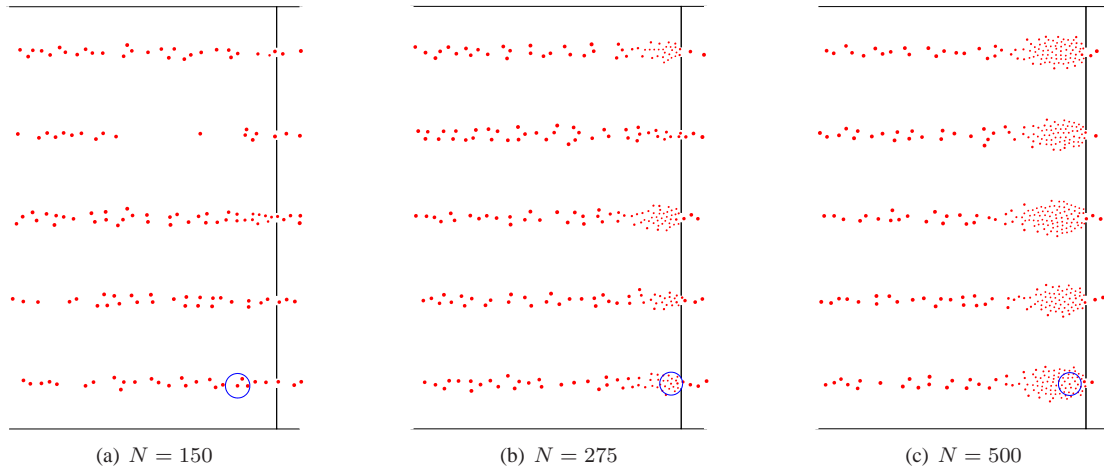


Abbildung 6.9: Momentaufnahme der Simulationen im inhomogenen Szenario nach 100 s Echtzeit für verschiedene Personenzahlen. Bei diesen Simulationen wurden die Brute-Force-Methode verwendet.

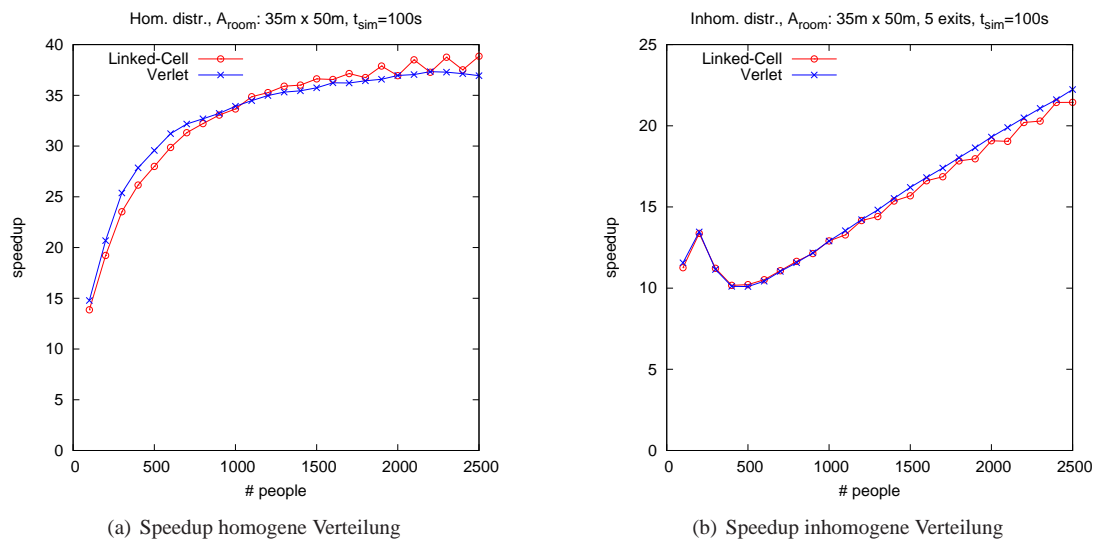


Abbildung 6.10: Vergleich des Speedups bei homogener und inhomogener Verteilung mit gleicher durchschnittlicher Dichte.

Die Laufzeit (hier nicht abgebildet) ist für die Brute-Force-Methode in beiden Szenarien nahezu identisch, da die Anzahl der Personen gleich ist. Aufgrund des höheren Speedups im homogenen Fall werden dort durch den Einsatz der Nachbarschaftslisten niedrigere Laufzeiten erreicht.

6.4 Resultate

Die Testszenarien zeigen, dass im Vergleich zu der Brute-Force-Methode mit Verlet- und Linked-Cell-Listen ein deutlicher Performancegewinn zu erzielen ist. Weiterhin haben sich die Überlegungen bestätigt, dass der Hauptaufwand einer Simulation die Berechnung der wechselwirkenden Kräfte ist. Darüber hinaus konnte gezeigt werden, dass für hohe Dichten bzw. große Personenzahlen der Speedup der Laufzeit entsprechend der notwendigen Kraftberechnungen skaliert (s. Kapitel 6.2.3).

Der Speedup hängt stark von verschiedenen Gegebenheiten ab, wie der Dichte und der Personenzahl. Dabei wurde bei einer homogenen Verteilung deutlich, dass eine höhere Dichte einen extrem geringen Speedup verursacht (für 1000 Personen sank der Speedup von ca. 37 bei 0,5 Personen/m² auf ca. 6 bei 5 Personen/m²). Bei konstanter Dichte hingegen ergibt sich mit wachsender Personenzahl das erwartete Skalierungsverhalten der Laufzeit von $\mathcal{O}(N^2)$ mit sehr kleinem Vorfaktor bei den Verlet-Listen und $\mathcal{O}(N)$ bei den Linked-Cell-Listen, wodurch der Speedup stetig mit N wächst. Bei einer Dichte von 3 Personen/m² ist eine Simulation mit Verlet-Listen für $N < 4000$ minimal schneller, für $N > 5000$ ergeben sich bei den Linked-Cell-Listen deutlich höhere Speedup-Werte. Interessant ist das Testszenario mit fester Raumgröße, bei dem der Speedup konvergiert und auch hier bei den Linked-Cell-Listen für große N höhere Werte als bei den Verlet-Listen annimmt. Die Laufzeit wächst dabei superlinear.

Das inhomogene Testszenario, das im Hinblick auf Evakuierungen von Interesse ist, zeigt ein Laufzeit- bzw. Speedup-Verhalten ähnlich dem homogenen Testfall mit konstanter Dichte. Aufgrund der Staubereiche an Engstellen kommt es zu einer quasi-konstanten Dichte, sodass der Speedup linear skaliert. Allerdings ist der Speedup wegen der lokal hohen Dichte deutlich geringer als in einem vergleichbaren homogenen Szenario. Die Ergebnisse aus dem homogenen Testfall lassen darauf schließen, dass das lineare Wachstum des Speedups im inhomogenen Fall beschränkt ist.

Insgesamt lässt sich sagen, dass die Linked-Cell-Listen bedingt durch die Gesamtkomplexität von $\mathcal{O}(N)$ insbesondere für große N einen größeren Performancegewinn als die Verlet-Listen liefern. Beispielsweise ist eine Simulation mit 10 000 Personen und einer Dichte von 3 Personen/m² mithilfe der Linked-Cell-Listen nahezu 70 mal, mit den Verlet-Listen ca. 60 mal schneller als mit der Brute-Force-Methode. Dahingegen eignen sich Verlet-Listen besser für hohe Dichten. Der Unterschied des Speedups beider Nachbarschaftslisten bei Simulationen des inhomogenen Testszenarios ist daher gering, wobei hier nur bis 2500 Personen simuliert wurde, sodass eine größere Abweichung für mehr

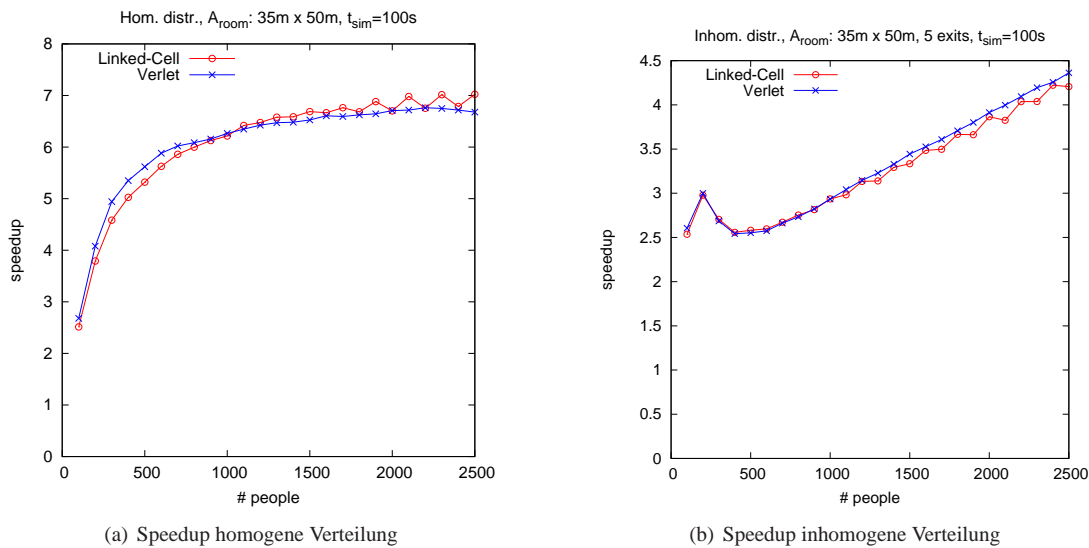


Abbildung 6.11: Ein geringerer Speedup resultiert aus einer anderen Definition der Brute-Force-Methode.

Personen erwartet wird.

Hinzuzufügen ist, dass bei Evakuierungen deutlich geringere Simulationslaufzeiten zu erwarten sind als in einem vergleichbaren Testszenario. Das ist damit zu begründen, dass die Personen nach Verlassen des Simulationsgebiets nicht weiter berücksichtigt werden und somit die Anzahl der Kraftberechnungen stetig abnimmt. Bei dem Anwendungsbeispiel im nächsten Kapitel wird dies ersichtlich.

Definition Brute-Force-Methode

Der Begriff Brute-Force-Methode lässt unterschiedliche Definitionen zu. Im Rahmen dieser Arbeit wird mit Brute-Force der intuitive, triviale Ansatz beschrieben, unabhängig vom Abstand die wechselwirkenden Kräfte zwischen allen Personen des Systems zu berechnen, da zunächst nicht klar ist, ob es sich um kurz- oder langreichweitige Kräftewirkungen handelt. Dieser Ansatz führt sowohl bei den Abstands- als auch bei den Kraftberechnungen zu einer Komplexität von $\mathcal{O}(N^2)$.

Im vorliegenden, konkreten Fall ist es aufgrund der kurzreichweitigen Kräfte allerdings möglich, wie auch bei den Nachbarschaftslisten in der Brute-Force-Methode die Kraft nur dann zu berechnen, wenn der Abstand zweier Fußgänger kleiner als der Interaktions-Abstand ist. Die Anzahl der Kraftberechnungen ist somit identisch, und der Speedup der Nachbarschaftslisten hängt allein von der Anzahl der Abstandsberechnungen ab.

Exemplarisch wurden mit dieser modifizierten Brute-Force-Methode die Simulationen für den homogenen Testfall mit konstanter Raumgröße (s. Kapitel 6.2.3) und den inho-

mogenen Testfall (s. Kapitel 6.3) wiederholt. Der jeweilige Speedup ist in Abbildung 6.11 dargestellt. Im Vergleich zu den vorherigen Ergebnissen ist erkennbar, dass der Speedup durch die Verwendung der modifizierten Brute-Force-Methode deutlich geringer ist, also die Laufzeit stark reduziert werden konnte. Durch die Verwendung von Nachbarschaftslisten ist jedoch noch immer ein Performancegewinn von 6-7 bzw. 4-4,5 gegenüber dieser Definition der Brute-Force-Methode zu verzeichnen.

Kapitel 7

Anwendungsbeispiel

Um eine erste Abschätzung für die Laufzeit eines realen Testszenarios in der *ESPRIT arena* zu erhalten, wird der im Projekt Hermes zu simulierende Bereich der Arena vereinfacht nachgebildet. In Abbildung 7.1, in der Momentaufnahmen der Simulation dargestellt sind, ist der Aufbau des Szenarios zu erkennen. Der betrachtete Bereich der *ESPRIT arena* beinhaltet eine Fankurve. Da mit der bisherigen Implementierung keine runden oder schrägen Wände modelliert werden können, werden die Abstände zwischen den Ausgängen so gut wie möglich approximiert. Betrachtet werden 12 Mundlöcher mit einer Breite von 2,4 m und einer Länge von 9 m, sowie 24 Ausgänge, die ebenfalls 2,4 m breit sind. Der Abstand zwischen den Mundlöchern und den Ausgängen wurde so berechnet, dass die Größe der entsprechenden Fläche mit dem realen Bereich übereinstimmt und beträgt 24 m. Der betrachtete Bereich ist für maximal 8200 Personen vorgesehen. Zusätzlich werden Simulationen mit 1000 und 4000 Personen durchgeführt. Für die Simulation werden die Personen in einem Bereich vor den Mundlöchern zufällig positioniert (s. Abbildung 7.1 oben). Dieser ist so dimensioniert, dass die durchschnittliche Dichte bei 8200 Personen ca. 1,9 Personen/m² beträgt.

Bei der Laufzeitmessung der Simulation ist zu beachten, dass es sich im Gegensatz zu dem inhomogenen Testszenario in Kapitel 6.3 nicht um ein periodisch geschlossenes System handelt, sondern um ein offenes Evakuierungsszenario. Das bedeutet, dass die Fußgänger nach einem hier definierten Bereich von 10 m hinter den Ausgängen das System verlassen und nicht mehr berücksichtigt werden müssen. Dadurch reduziert sich im Laufe der Simulation die Anzahl der simulierten Personen und Kraftberechnungen. Die Simulation wird solange durchgeführt, bis alle Fußgänger das System verlassen haben, was der Evakuierungszeit entspricht.

In Tabelle 7.1 sind die gemessenen Laufzeiten der Simulationen mithilfe der Nachbarschaftslisten für verschiedene Personenzahlen aufgelistet. Zusätzlich ist die simulierte Echtzeit ($n_t \cdot \Delta t$, n_t : Anzahl Zeitschritte bis der letzte Fußgänger das System verlässt) angegeben. Aus dem Quotient der Echtzeit und der Laufzeit ergibt sich jeweils der Be-

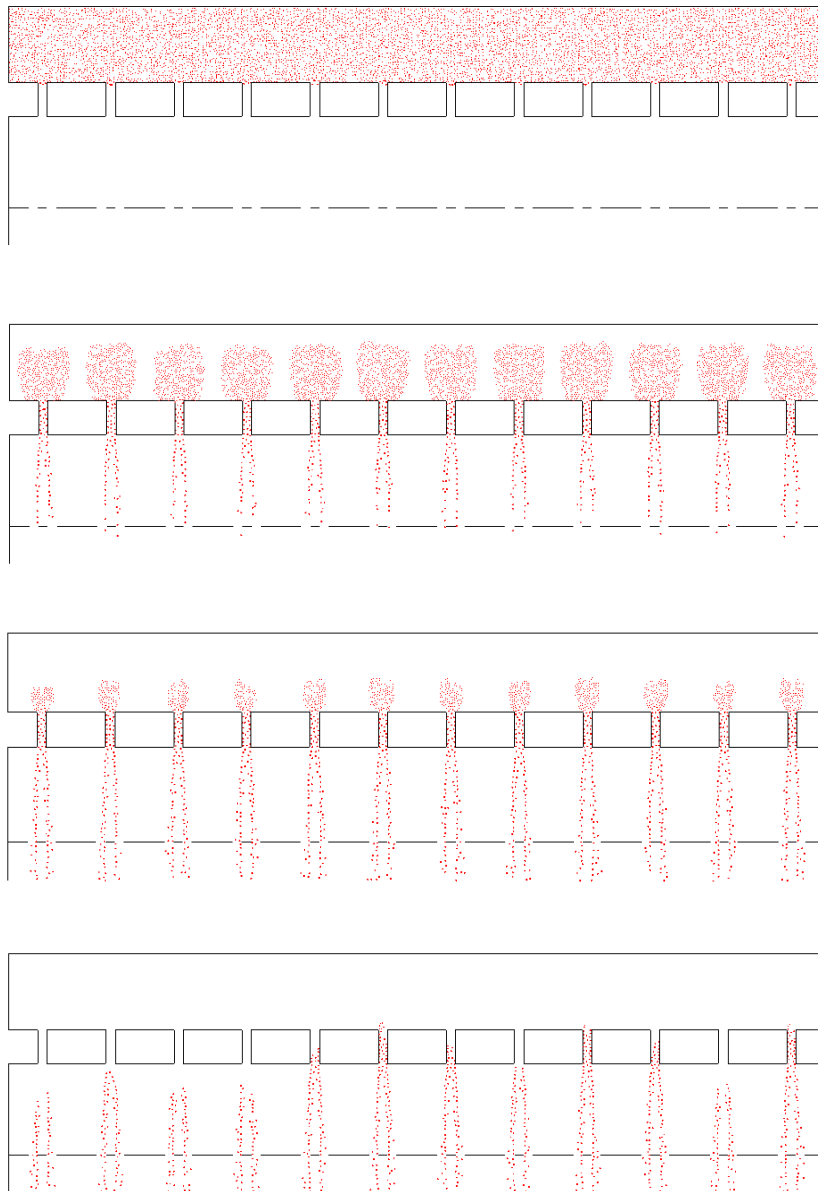


Abbildung 7.1: Momentaufnahmen der Simulation mit 8200 Personen. Die Personen werden im Startbereich vor den Mundlöchern zufällig positioniert und verlassen daraufhin das Simulationsgebiet durch die Mundlöcher und die Ausgänge.

Anzahl Personen	Linked-Cell-Listen	Verlet-Listen	Echtzeit	Beschleunigungsfaktor (Linked-Cell)	Beschleunigungsfaktor (Verlet)
1000	20,46 s	23,66 s	75,61 s	3,70	3,20
4000	386,41 s	468,91 s	172,05 s	0,45	0,37
8200	1843,90 s	2433,14 s	303,42 s	0,16	0,12

Tabelle 7.1: Laufzeiten der Simulation eines Szenarios, das einen Bereich der ESPRIT arena approximiert, für verschiedene Personenzahlen unter der Verwendungen von Linked-Cell- und Verlet-Listen. Zusätzlich wird die Evakuierungszeit (Echtzeit der Simulation) und der jeweilige Beschleunigungsfaktor aufgelistet, der sich aus der Division der Echtzeit durch die Laufzeit ergibt.

schleunigungsfaktor.

Mit steigender Personenzahl und die dadurch verursachten Stauungen entstehen neben einer erhöhten Evakuierungszeit ebenfalls mehr Kraftberechnungen und dadurch längere Laufzeiten. Am Beschleunigungsfaktor ist zu erkennen, dass die Simulationen für 4000 und 8200 Personen ca. 2 bis 8 mal länger als Echtzeit benötigen, wohingegen für 1000 Personen eine Berechnung schneller als das 3,7-fache (Linked-Cell) bzw. 3,2-fache (Verlet) als Echtzeit ermöglicht wird.

Um die maximale Anzahl von Personen in Echtzeit zu simulieren, muss die Performance um einen weiteren Faktor 10 gesteigert werden. Allerdings ist dies eine grobe Abschätzung mithilfe eines vereinfachten Szenarios. Es ist daher zu vermuten, dass ein größerer Performancegewinn als 10 erzielt werden muss.

Kapitel 8

Ausblick

Das Anwendungsbeispiel verdeutlicht, dass weitere Optimierungsmöglichkeiten ausgenutzt werden müssen, um das in Hermes angestrebte Ziel einer Simulation schneller als Echtzeit zu erreichen. Dabei können zur Performancesteigerung sowohl verbesserte Techniken und neue Konzepte für Nachbarschaftslisten als auch hardware-orientierte Algorithmen eingesetzt werden.

Des Weiteren kann die Laufzeit mithilfe einer Parallelisierung des Programms zusätzlich reduziert werden. Besonders bei großen Personenzahlen kann die Effizienz mit Parallelisierungstechniken gesteigert werden. Der Schwerpunkt sollte dabei auf den Berechnungen der interaktiven Wechselwirkungen zwischen Personen liegen.

Neben der Laufzeitoptimierung sind zudem Weiterentwicklungen in der Modellierung zur Verbesserung der Dynamik der Fußgängerströme nötig. Insbesondere muss für eine Simulation von Szenarien mit komplexen Geometriestrukturen die Interaktion von Fußgängern mit ihrer Umgebung untersucht und modelliert werden.

8.1 Nachbarschaftslisten

Für eine zusätzliche, mögliche Reduzierung der Laufzeit können die im Rahmen dieser Arbeit implementierten Nachbarschaftslisten miteinander kombiniert werden. Hierbei wird für die Verwaltung der Nachbarn die Verlet-Methode verwendet, wobei die Nachbarschaftslisten nach dem Linked-Cell-Verfahren aktualisiert werden, indem nicht zu allen, sondern nur zu den Personen der Nachbarzellen der Abstand berechnet wird. Durch diese Kombination wird die Gesamtkomplexität der Verlet-Listen von $\mathcal{O}(N^2)$ auf $\mathcal{O}(N)$ reduziert.

Weiterhin existieren Ansätze, die sich mit der Problematik der indirekten Adressierung der Personen bzw. Teilchen eines Systems beschäftigen, durch die häufige Cache-Misses

und daher eine geringere Performance resultieren. Um dem entgegenzuwirken, kann eine Sortierung der Personen erfolgen, sodass im Simulationsgebiet benachbarte Personen auch in benachbarten Speichern liegen. In [20] wurde gezeigt, dass hierdurch bei den Linked-Cell-Listen ein Performancegewinn von Faktor 2 bis 3 erreicht werden kann, da die Cache Hit Rate vergrößert wurde. Bei den Verlet-Listen wird in [7] für eine effektivere Kraftberechnung das Pipelining möglichst optimal ausgenutzt, wodurch eine Reduzierung der Rechenzeit erreicht wird.

Speziell für das hier implementierte Modell könnte der Rechenaufwand verringert werden, wenn bei den Nachbarschaftslisten der Blickwinkel der Fußgänger berücksichtigt wird. Ähnlich wie bei der Kraftberechnung im CFM (s. Kapitel 3.3.1) könnte dadurch die Anzahl der gespeicherten bzw. betrachteten Nachbarn auf nur diejenigen beschränkt werden, die für die Kraftberechnung relevant sind. Dies würde sowohl die Anzahl der Abstands- als auch der Kraftberechnungen reduzieren und bei den Verlet-Listen zusätzlich den Speicherbedarf halbieren.

Weitere Untersuchungen müssen zudem klären, ob beim Verfahren der Linked-Cell-Listen durch Variation der Zellgröße zusätzlicher Performancegewinn verzeichnet werden kann. Insbesondere bei hohen Personendichten werden Zellen mit einer Kantenlänge kleiner als R_c eine positive Auswirkung auf die Rechenzeit haben.

8.2 Parallelisierung

Im Rahmen des Projekts Hermes soll das Simulationsprogramm für Cell-Prozessoren geeignet parallelisiert werden [10]. Die Struktur eines Cell-Prozessors ([3]) im Hinblick auf die Speicherverwaltung ist in Abbildung 8.1 dargestellt. Der Prozessor, der von IBM gemeinsam mit Sony und Toshiba entwickelt wurde, besteht aus einem *PowerPC Processing Element* (PPE) und acht sogenannten *Synergistic Processing Elements* (SPE). PPE, SPEs und Ein- und Ausgang sind auf dem Cell-Chip durch einen speziellen Datenbus mit hoher Bandbreite, dem *Element Interconnect Bus* (EIB), verbunden. Mittels DMA werden über den EIB Daten zwischen dem lokalen Speicher und dem Hauptspeicher transferiert.

Jedes SPE setzt sich zusammen aus einer *Synergistic Processing Unit* (SPU), der Recheninheit, einem *Memory Flow Controller* (MFC) zur Steuerung der DMA-Übertragungen, sowie einem lokalen Speicher der Größe 256 KB. Lade- und Speicheroperationen können nur auf dem lokalen Speicher durchgeführt werden, für den Zugriff auf den Hauptspeicher muss DMA verwendet werden. Hervorzuheben ist, dass die SPUs Vektorprozessoren mit vierfachem SIMD sind, da sie hauptsächlich für Bild- und Videoverarbeitung ausgelegt wurden, sodass bei entsprechenden Datenstrukturen und Operationen eine hohe Performance erzielt werden kann.

Das PPE ist die Steuereinheit des Cell-Prozessors und für diese Zwecke optimiert. Es basiert auf der 64-Bit-PowerPC-Architektur von IBM. Mithilfe des PPEs können Prozesse

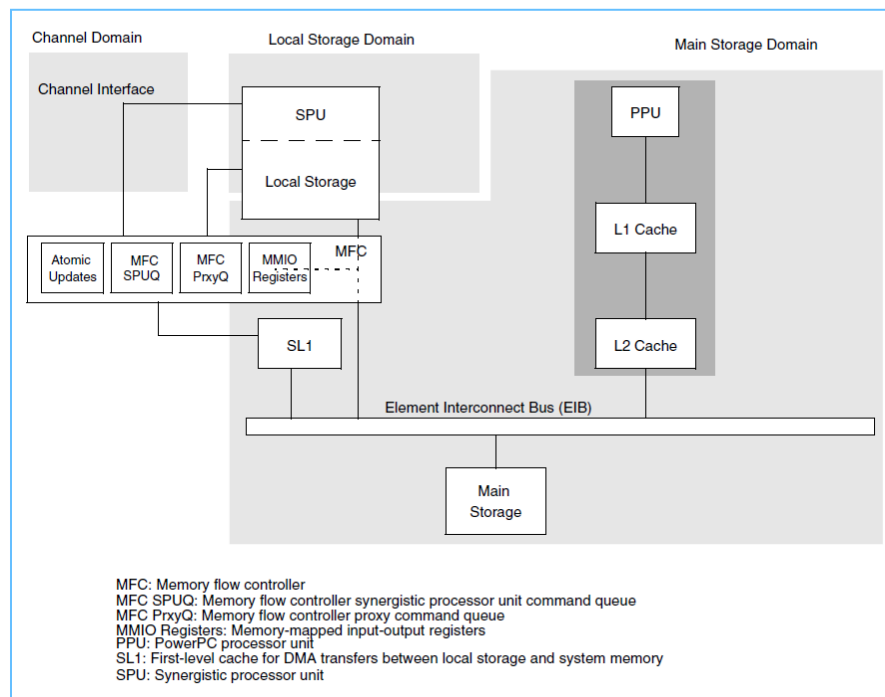


Abbildung 8.1: Aufbau des Cell-Prozessors mit Priorität auf die Speicherverwaltung. (Quelle: IBM)

auf den SPEs gestartet, gestoppt, unterbrochen oder terminiert werden. Im Gegensatz zu den SPEs kann das PPE mit normalen Lese- und Schreiboperationen sowohl auf den Hauptspeicher als auch auf den lokalen Speicher der SPEs zugreifen.

Die Hauptrechenleistung wird durch die SPEs bestimmt. Allerdings stellen die Verwendung von DMA zum Transfer der Daten und der begrenzte lokale Speicher von 256 KB für Software-Entwickler eine große Herausforderung dar.

Bei einer Parallelisierung ist neben dem für sequentielle Programme relevanten Rechen- und Speicherbedarf zusätzliche Kommunikation nötig. Dabei gilt es, den Kommunikationsaufwand so gering wie möglich zu halten. Weiterhin sollte der serielle Anteil, der in jedem parallelen Programm vorhanden ist, minimiert werden, da es zu einer von ihm abhängigen Sättigung des Speedups kommt (Amdahlsche Gesetz). Ein große Rolle bei der Parallelisierung spielt die Lastverteilung. Die Rechenlast sollte möglichst homogen auf alle Prozessoren verteilt werden, sodass es nicht zu unnötigen Wartezeiten und damit zu einer längeren Laufzeit und ungenutzter Rechenkapazität kommt.

Es existieren unterschiedliche Strategien, um ein Problem zu parallelisieren. Bei der Gebietszerlegung wird der betrachtete Bereich in Teile entsprechend der Anzahl der Prozessoren segmentiert, sodass jeder Prozessor das ihm zugeordnete Teilgebiet bearbeitet. Üblicherweise besteht eine Relation zwischen benachbarten Gebieten, sodass in regelmäßigen

Zeitintervallen ein Austausch an Informationen bei den jeweils angrenzenden Rändern der Teilgebiete erfolgen muss. Aufgrund komplexer Geometriestrukturen (hier die *ES-PRIT arena*) ist diese Möglichkeit zur Parallelisierung eher ungeeignet. Hinzu kommt, dass die Verteilung der Fußgänger meist inhomogen ist und dadurch eine gleichmäßige Lastverteilung schwierig oder kaum möglich ist.

Ein weiterer Ansatz ist die Parallelisierung nach dem Master-Slave-Prinzip, bei dem das Problem in viele Teilprobleme unterteilt wird. Der Master-Prozess ist dafür zuständig, die einzelnen Teilprobleme so an die Slave-Prozesse zu verteilen, dass zu jeder Zeit möglichst alle Slave-Prozesse rechnen. Seine Aufgabe ist, die jeweiligen Resultate der Slave-Prozesse einzusammeln und zu einem Gesamtergebnis zusammenzufügen. Ein Slave-Prozess wartet ununterbrochen darauf, dass ihm der Master-Prozess eine Aufgabe schickt, woraufhin er diese bearbeitet, das Ergebnis an den Master-Prozess übermittelt und anschließend erneut wartet.

Der Vorteil bei diesem Prinzip ist eine homogene Lastverteilung, insbesondere bei Inhomogenitäten und ausreichend hoher Anzahl von Teilproblemen. Allerdings ist auf einen möglichst minimalen Kommunikationsaufwand zu achten. Bei dem Cell-Prozessor kann das PPE die Rolle des Master-Prozesses einnehmen, wohingegen die SPEs die Slaves sind.

Es sind verschiedene Möglichkeiten denkbar, das Gesamtproblem der Simulation von Fußgängern in einer komplexen Geometriestruktur für eine Parallelisierung nach dem Master-Slave-Prinzip in Teilprobleme zu zerlegen:

- Eine Zerlegung könnte anhand der Räume erfolgen, sodass einem SPE immer ein Raum mit den entsprechenden Fußgängern zugeteilt wird. Allerdings kann es zu einem Ungleichgewicht bezüglich der Lastverteilung kommen, wenn es einige große Räume mit vielen Fußgängern und kleine Räume mit wenigen oder gar keinen Fußgängern gibt. Weiterhin ist zu beachten, dass die Anzahl der Teilprobleme und damit der Räume für eine homogene Lastverteilung ausreichend groß sein muss, insbesondere bei unterschiedlich stark besetzten Räumen. Darüber hinaus könnte der Kommunikationsaufwand zu groß sein, da die Raum- und Fußgängerinformationen in jedem Zeitschritt an die SPEs übermittelt werden müssen.
- Eine weitere Möglichkeit ergibt sich bei der Verwendung von Linked-Cell-Listen durch die Unterteilung der gesamten Geometriestruktur in Zellen. Ähnlich dem Prinzip der Gebietszerlegung könnten mehrere Zellen als Teilgebiet definiert werden und den SPEs zugeordnet werden. Hierbei müssten für die komplette Kraftberechnung neben den Fußgängerdaten auch die zugehörigen Raumdaten (unter Umständen von mehreren Räumen) übermittelt werden. Wie bei der ersten Möglichkeit ist auch hier ein Austausch der Daten in jedem Zeitschritt notwendig.
- Einen möglicherweise geringeren Aufwand könnte durch eine Variante entstehen, bei der nur die interaktiven Kraftberechnungen auf die SPEs verlagert wird. Hierfür

müssten die Daten einer Person und der für die Wechselwirkungen relevanten Nachbarn übermittelt werden. Durch geschickte Ansätze könnten zudem für mehrere benachbarte Personen die interaktiven Kräfte berechnet werden, sodass insgesamt weniger Daten übertragen werden müssen und sich das Verhältnis von Berechnung zu Kommunikation verbessert.

Bei allen aufgezählten Möglichkeiten besteht das Problem, dass die relevanten Daten in jedem Zeitschritt übertragen werden müssen, sodass der nötige Kommunikationsaufwand die Gesamtperformance beeinträchtigen könnte. Für entsprechende Abschätzungen sind zunächst Messungen bezüglich des Datenaustauschs auf dem Cell-Prozessor erforderlich.

Um einen Transfer von Daten in jedem Zeitschritt zu vermeiden, müssen die SPEs mit unabhängigen Daten rechnen. Unabhängig bedeutet hierbei, dass die Fußgänger, die auf einem SPE simuliert werden, für eine gewisse Zeit nicht mit Fußgängern der anderen SPEs interagieren. Beispielsweise könnte es sich um verschiedene Räume handeln. Die Schwierigkeit besteht darin, die Fußgänger bzw. die Geometrie auf die acht vorhandenen SPEs so aufzuteilen, dass

- eine homogene Lastverteilung gewährleistet ist,
- die Zeitpunkte der zukünftigen Interaktionen möglichst übereinstimmen,
- die Zeitpunkte der zukünftigen Interaktionen möglichst einfach bestimmt werden können.

Bezüglich der Parallelisierung des Simulationsprogramms muss der bereits weiter oben erwähnte begrenzte lokale Speicher der SPEs berücksichtigt werden. Durch die Speicherlimitierung ist nur eine Simulation einer beschränkten Anzahl von Fußgängern auf einem SPE möglich. Werden auf einem SPE Nachbarschaftslisten verwendet, muss der zusätzlich benötigte Speicher (besonders bei den Verlet-Listen) beachtet werden.

Für eine erste Einschätzung des Platzbedarfs wurde das gesamte Simulationsprogramm auf einem SPE ausgeführt und für die Brute-Force-Methode, die Linked-Cell-Listen und die Verlet-Listen die maximal mögliche Anzahl von Personen ermittelt. Da die Simulationsdauer keinen Einfluss auf den Speicher hat, wurde hier jeweils für 10 s simuliert. Für die Simulation wurde das homogene Testszenario (s. Kapitel 6.2) mit einer Raumgröße von $20\text{ m} \times 20\text{ m}$ verwendet. Zusätzlich wurde für die Simulation mit der Brute-Force-Methode und den zwei Nachbarschaftslisten die benötigte Rechenzeit mit der Personenzahl gemessen, die mit den Verlet-Listen maximal möglich ist. Die jeweiligen maximalen Personenzahlen und Laufzeiten sind in Tabelle 8.1 aufgelistet.

Wie schon vermutet, ist die maximal mögliche Personenzahl bei Verwendung von Verlet-Listen deutlich niedriger als bei der Brute-Force-Methode oder bei den Linked-Cell-Listen. Für die Linked-Cell-Listen ist der Speicherbedarf nicht wesentlich größer als für die Brute-Force-Methode. Bemerkenswert sind die Laufzeiten der Simulationen mit

	Brute-Force	Linked-Cell-Listen	Verlet-Listen
max. Personenzahl	809	768	166
Laufzeit mit 166 Personen	14,9 s	1,6 s	21,52 s

Tabelle 8.1: Maximal mögliche Personenzahlen und Laufzeiten zur Simulation mit 166 Personen auf einem SPE bei Verwendung der Brute-Force-Methode, der Linked-Cell- und der Verlet-Listen. Dabei wurde das komplette Programm auf einem SPE ausgeführt.

166 Personen. Linked-Cell-Listen beschleunigen die Simulation im Vergleich zur Brute-Force-Methode erheblich, wohingegen die Verlet-Listen sogar für eine langsamere Laufzeit sorgen. Genaue Überprüfungen müssen auf dem Cell-Prozessor erfolgen, um die Ursachen zu klären und die Algorithmen der entsprechenden Architektur optimal anzupassen. Weiterhin ist die Effizienz der Nachbarschaftslisten für größere Personenzahlen höher, sodass eine Verwendung von Nachbarschaftslisten eventuell auf das PPE beschränkt werden sollte. Konkrete Tests auf dem Cell-Prozessor werden jedoch nötig sein, um letztendlich eine Aussage machen zu können, welches die beste Parallelisierungsstrategie und damit auch die beste Verwendung von Nachbarschaftslisten ist.

8.3 Modellierung

Aus dem in Kapitel 3.3.1 eingeführten Modell resultiert ein System gekoppelter Differentialgleichungen, welches durch ein numerisches Verfahren in diskreten Zeitschritten gelöst wird. Die Größe des Zeitschrittes hängt linear mit der Rechenzeit zusammen. Eine mögliche Optimierung kann der Einsatz anderer numerischer Verfahren bieten, sodass das Zeitintervall erhöht und so die Rechenzeit verringert werden kann. Denkbar ist hier ein Lösungsverfahren (Störmer-Verlet-Verfahren), das im Bereich der Molekulardynamik für Hamiltonsche Systeme eingesetzt wird und möglicherweise größere Zeitschritte erlaubt.

Weiterhin ist bei der Modellierung unklar, wie die abstoßenden Kräfte formuliert werden können. In Bezug auf die Verwendung von Nachbarschaftslisten ist der Cutoff-Radius bei der Kräfterdefinition von großer Bedeutung. Denn je kleiner der Cutoff-Radius gewählt werden kann, desto effektiver ist der Einsatz von Nachbarschaftslisten. Weiterhin reduziert sich hiermit der Speicherbedarf bei Verwendung von Verlet-Listen.

Für die Simulation realer Szenarien muss die Interaktion der Fußgänger mit komplexen Raumstrukturen, wie Hindernissen, runden oder schrägen Wänden oder Ecken modelliert werden. Ein Aspekt dabei ist die Wahl des Zielpunktes bei Ausgängen. Ein Vorschlag, der zu einer weiteren Verbesserung führen könnte, ist, den Zielpunkt um $D_i/2$ nach „innen“ zu verlagern. Denn anderenfalls erfährt eine Person, die parallel entlang einer Wand zum Ausgang läuft, ständig eine von dieser Wand abstoßende Kraft, da die Person versucht, schräg zum Mittelpunkt des Ausgangs zu laufen (s. Abbildung 8.2(a)). Aufgrund des Volumenausschlusses und der relativen Geschwindigkeit (s. Gleichung (3.3)) wirkt

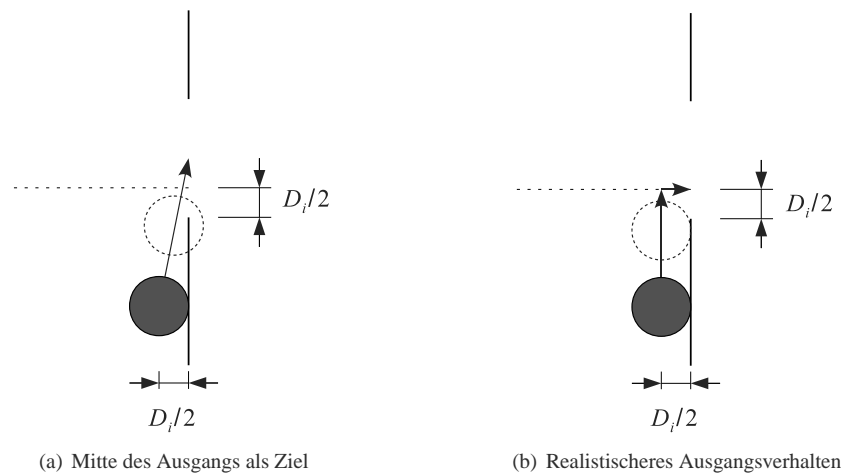


Abbildung 8.2: In (a) ist die Wunschrichtung des Fußgängers bei Festlegung des Zielpunktes auf die Mitte des Ausgangs dargestellt. Der Fußgänger erfährt während des Wegs zum Ausgang ständig abstoßende Kräfte von der Wand. Zur Verdeutlichung ist die Position des Fußgängers bei nicht vorhandenen abstoßenden Kräften visualisiert (gestrichelt), wodurch es zu einer Überlappung mit der Wand käme. (b) zeigt die Wunschrichtungen, nach denen sich ein Fußgänger in der Realität bewegt. Er läuft bis zum Ausgang parallel zur Wand und erfährt wegen Gleichung (3.3) keine abstoßenden Kräfte.

von der Wand eine Kraft auf die Person, sodass die Fortbewegung gebremst wird und eine womöglich schlechtere Einfädlung am Ausgang in den Personenstrom erfolgt. Realistischer scheint zu sein, dass ein Fußgänger parallel zur Wand läuft und erst in Höhe des Ausgangs die entsprechende Richtung einschlägt (s. Abbildung 8.2(b)).

Kapitel 9

Zusammenfassung

Aufgrund einer steigenden Anzahl von Großveranstaltungen und einer wachsenden Komplexität der Gebäude spielt für die Sicherheit der Personen besonders die günstige und schnelle Evakuierung eine wichtige Rolle. Im Rahmen des Projekts Hermes wird am Beispiel der *ESPRIT arena* ein Evakuierungsassistent erforscht, der anhand der aktuellen Situation innerhalb einer Minute eine Prognose der Fußgängerbewegungen der nächsten 15 Minuten liefern soll. Da bis zu 60 000 Fußgänger simuliert werden sollen, stellt dies besondere Ansprüche an die Rechenperformance. Die vorliegende Arbeit steuert dem Ziel bei, eine Simulation schneller als Echtzeit zu ermöglichen. Hierfür wurde ein Programm zur Simulation raumkontinuierlicher Modelle entwickelt. Aufgrund der kurzreichweitigen Kräfte können zur Laufzeitoptimierung Nachbarschaftslisten eingesetzt werden, die bereits im Bereich der Molekulardynamik erfolgreich verwendet werden.

In dieser Arbeit wurden zwei Ansätze, die Verlet- und die Linked-Cell-Listen, implementiert und ihre Performance im Vergleich zur Brute-Force-Methode analysiert. Für die Tests wurde zwischen homogenen und inhomogenen Szenarien unterschieden und der Einfluss von Dichte und Personenzahl untersucht.

Bei dem homogenen Testszenario wurde festgestellt, dass gemäß den Erwartungen der Speedup mit steigender Dichte abnimmt. Bei einer Simulation von 1000 Personen und einer Variation der Dichte von 0,5 bis 5 Personen/m² reduziert sich der Speedup um ca. 85 %. Weiterhin wurde festgestellt, dass es in einem Szenario mit fester Raumgröße und wachsender Personenzahl zu einer Konvergenz des Speedups kommt. Dies ist im Hinblick auf spätere Anwendungen interessant, bei denen die Raumgröße vorgegeben ist. Ferner wächst der Speedup bei konstanter Dichte mit der Anzahl der Personen, wobei das Wachstum bei den Linked-Cell-Listen linear, bei den Verlet-Listen wegen der $\mathcal{O}(N^2)$ -Komplexität schwächer als linear ist.

Bei einer inhomogenen Verteilung ergibt sich aufgrund der Staubereiche eine quasi-konstante Dichte und dadurch ähnlich dem homogenen Testfall eine scheinbar lineare Stei-

gerung des Speedups. Bei größeren Personenzahlen ist auch hier eine Konvergenz des Speedups zu erwarten. Der Vergleich eines homogenen und inhomogenen Szenarios zeigt, dass der Speedup im homogenen Fall schnell gegen den Grenzwert konvergiert und deutlich höher ist. Begründet ist dies mit der Homogenität der Fußgängerverteilung und der damit verbundenen lokal niedrigen Dichte.

Insgesamt wird durch die Verwendung von Nachbarschaftslisten in den Testszenarien ein Performancegewinn bis zu Faktor 70 erreicht. Dabei ist mithilfe der Linked-Cell-Listen für große Personenzahlen ein zum Teil deutlich höherer Speedup als mit den Verlet-Listen möglich, bei höheren Dichten ist der Unterschied zu Verlet-Listen geringer.

Mit der Simulation eines Beispielszenarios, das den im Projekt Hermes betrachteten Bereich der *ESPRIT arena* repräsentiert, wurde gezeigt, dass bei dem verwendeten raumkontinuierlichen Modell weitere Optimierungsmöglichkeiten erforderlich sind, um eine Simulation schneller als Echtzeit zu erreichen. Folgende Ansätze können hierfür verfolgt werden:

- eine Kombination der beiden Nachbarschaftslisten,
- eine Sortierung der Listen bzgl. der Lokalität der Personen,
- spezielle auf das Pipelining ausgerichtete Algorithmen,
- Vereinfachungen des Fußgängermodells, beispielsweise durch Vergrößern des Zeitschrittes zum Lösen des Systems von Differentialgleichungen oder durch Verringern des maximalen Interaktions-Abstands,
- Parallelisierungstechniken.

Im Rahmen des Projekts Hermes sollen die Simulationen auf einem Cell-Prozessor ausgeführt werden. Hierbei sind jedoch weitere technische Probleme zu lösen, z. B. der begrenzte lokale Speicher der einzelnen Prozessoren oder die Aufteilung des Problems in Teilprobleme.

Anhang A

Symbole und Abkürzungen

A.1 Symbole

ρ	Dichte in Personen pro Quadratmeter
ρ_{max}	maximal anzunehmende Dichte in Personen pro Quadratmeter
J	Fluss
J_s	spezifischer Fluss
b	Breite der Durchflussstelle
N	Anzahl der Personen
m_i	Masse der i -ten Person
τ	Relaxationskoeffizient zur Steuerung der Beschleunigung
$\vec{x}_i = (x_i, y_i)^T$	Position der i -ten Person
\vec{v}_i	aktuelle Geschwindigkeit der i -ten Person
\vec{v}_i^0	Wunschgeschwindigkeit und -richtung der i -ten Person
D_i	Durchmesser der i -ten Person
D_{min}	minimaler Durchmesser einer Person
D_{max}	maximaler Durchmesser einer Person

\vec{F}_i^{drv}	treibende Kraft, die auf die i -te Person wirkt
\vec{F}_i^{rep}	gesamte abstoßende Kraft, die auf die i -te Person wirkt
\vec{f}_{ij}^{rep}	abstoßende Kraft zwischen i -ter und j -ter Person
\vec{f}_{iw}^{rep}	abstoßende Kraft zwischen i -ter und w -tem Hindernis
\vec{R}_{ij}	Differenzvektor zwischen den Positionen der i -ten und j -ten Person
v_{ij}	relative Geschwindigkeit zwischen i -ter und j -ter Person
\vec{e}_{ij}	Richtungsvektor von i -ter zu j -ter Person
\vec{e}_{iw}	Richtungsvektor von i -ter Person zu w -tem Hindernis
K_{ij}, K_{iw}	Koeffizient zur Berücksichtigung des Blickwinkels der i -ten Person
$e_x^1, e_y^1, e_x^2, e_y^2$	Koordinaten eines Ausgangs
$E = E(D_i, D_j)$	effektiver (Interaktions-)Abstand zwischen i -ter und j -ter Person in Abhängigkeit von dem Platzbedarf der Personen
E_{max}	maximaler effektiver Abstand, bis zu dem eine Kraft wirkt
N^+	Anzahl der betrachteten/gespeicherten Nachbarn
N^*	Anzahl der Nachbarn, die in die Kraftberechnung einfließen
R_c	Cutoff-Radius bei den Nachbarschaftslisten
R_s	Skin-Radius bei den Verlet-Listen
L_c	Breite der Zellen bei den Linked-Cell-Listen
n	Anzahl der Zeitschritte bis zur nächsten Aktualisierung der Verlet-Listen
Δt	Größe der Zeitschritte

S	Speedup: Verhältnis der Laufzeit mit der Brute-Force-Methode zu der Laufzeit mit der jeweiligen Nachbarschaftsliste
t_{sim}	simulierte Echtzeit
A_{room}	Größe des Raumes bei den Testszenarien
t_{calc}	Rechenzeit

A.2 Abkürzungen

<i>BMBF</i>	Bundesministerium für Bildung und Forschung
<i>CA</i>	Zelluläre Automaten (engl. <i>Cellular Automata</i>)
<i>CDT</i>	Kollisionserkennung/Konfliktlösung (engl. <i>Collision Detection Technique</i>)
<i>CFM</i>	Zentrifugalkräfte-Modell (engl. <i>Centrifugal Force Model</i>)
<i>DMA</i>	Speicherdirektzugriff (engl. <i>Direct Memory Access</i>)
<i>EIB</i>	Element Interconnect Bus
<i>MFC</i>	Memory Flow Controller
<i>OpenGL</i>	Open Graphics Library
<i>PPE</i>	PowerPC Processing Element
<i>SIMD</i>	Single Instruction, Multiple Data
<i>SPE</i>	Synergistic Processing Element
<i>SPU</i>	Synergistic Processing Unit

Abbildungsverzeichnis

1.1	Prinzip des Evakuierungsassistenten	2
2.1	Fundamentaldiagramm	4
2.2	Linienbildung	5
3.1	Mögliche Nachbarschaften bei Zellulären Automaten	9
3.2	Kräfte mit relativen Geschwindigkeiten beim CFM	11
3.3	Kräfte mit Volumenausschluss beim CFM	12
3.4	Zielbestimmung eines Fußgängers	15
3.5	Verbesserte Variante der Zielbestimmung eines Fußgängers	16
3.6	Berücksichtigung des Sichtkontakts	16
4.1	Hauptroutine des Simulationsprogramms.	20
4.2	Zufällige Verteilung der Personen, Variante 1	21
4.3	Zufällige Verteilung der Personen, Variante 2	21
4.4	Visualisierung von Trajektorien mit dynamischer Farbwahl	22
5.1	Bestimmung des Cutoff-Radius	26
5.2	Nachbarschaftsumgebung bei Verlet-Listen	28
5.3	Skizze zum Skin-Radius	29
5.4	Listen-Aktualisierung bei Verlet-Listen	30
5.5	Kraftberechnung bei Verlet-Listen	30
5.6	Aufteilung des Simulationsgebiets bei Linked-Cell-Listen	31
5.7	Verwendete Datenstrukturen bei Linked-Cell-Listen	32
5.8	Listen-Aktualisierung bei Linked-Cell-Listen	33
5.9	Kraftberechnung bei Linked-Cell-Listen	34
5.10	Globale Gebietsaufteilung bei Linked-Cell-Listen	35
6.1	Laufzeitmessungen bei Verwendung der Verlet-Listen mit Variation des Skin-Radius R_s	38
6.2	Laufzeit und Speedup bei homogener Verteilung und konstanter Dichte in Abhängigkeit von der Personenzahl	40
6.3	Speedup bei homogener Verteilung und konstanter Personenzahl in Abhängigkeit von der Dichte	41
6.4	Speedup bei homogener Verteilung und fester Raumgröße in Abhängigkeit von der Personenzahl	43
6.5	Laufzeit bei homogener Verteilung und fester Raumgröße in Abhängigkeit von der Personenzahl	44
6.6	Szenario für eine inhomogene Verteilung	45
6.7	Laufzeit und Speedup bei inhomogener Verteilung in Abhängigkeit von der Personenzahl	46

TABELLENVERZEICHNIS

6.8	Speedup bei inhomogener Verteilung für kleine Personenzahlen	47
6.9	Simulation im inhomogenen Szenario nach 100 s Echtzeit für verschiedene Personenzahlen	48
6.10	Vergleich des Speedups bei homogener und inhomogener Verteilung	48
6.11	Speedup bei modifizierter Brute-Force-Methode	50
7.1	Anwendungsbeispiel <i>ESPRIT arena</i>	54
8.1	Aufbau des Cell-Prozessors	59
8.2	Wahl des Ziels bei einem Ausgang	63

Tabellenverzeichnis

7.1	Laufzeit-Messergebnisse der Simulation der <i>ESPRIT arena</i>	55
8.1	Maximale Personenzahl auf einem SPE	62

Literaturverzeichnis

- [1] M. P. Allen and D. J. Tildesley. *Computer Simulation of Liquids*. Oxford Science Publications, Oxford, 1987.
- [2] V. J. Blue and J. L. Adler. Modeling Four-Directional Pedestrian Flows. *Transportation Research Record*, 1710:20–27, 2000.
- [3] Cell Broadband Engine Architecture. <http://www.ibm.com/developerworks/power/cell/>.
- [4] U. Chattaraj, A. Seyfried, and P. Chakroborty. Comparison of Pedestrian Fundamental Diagram Across Cultures. *Advances in Complex Systems (ACS)*, 12(3):393–405, 2009. arXiv:0903.0149.
- [5] M. Chraïbi, A. Seyfried, A. Schadschneider, and W. Mackens. Quantitative Description of Pedestrian Dynamics with a Force-based Model. 2009 IEEE/WIC/ACM International Joint Conferences on Web Intelligence and Intelligent Agent Technologies (WI-IAT 2009), to be held 15-18 September 2009, in Milano, Italy, 2009.
- [6] M. Fukui and Y. Ishibashi. Jamming Transition in Cellular Automaton Models for Pedestrian on Passageway. *Journal of the Physical Society of Japan*, 68(11):3738–3739, nov 1999.
- [7] S. Gupta. Computing aspects of molecular dynamics simulation. *Computer Physics Communications*, 70:243–270, 1992.
- [8] D. Helbing and P. Molnár. Social force model for pedestrian dynamics. *Phys. Rev. E*, 51:4282–4286, 1995.
- [9] R. W. Hockney and J. W. Eastwood. *Computer Simulation Using Particles*. McGraw-Hill, New York, 1981.
- [10] S. Holl and A. Seyfried. Hermes - an Evacuation Assistant for Mass Events. *inSide*, 7(1):60–61, 2009.
- [11] M. Muramatsu and T. Nagatani. Jamming transition in two-dimensional pedestrian traffic. *Physica A*, 275:281–291, 2000.
- [12] D. R. Parisi and C. O. Dorso. Microscopic dynamics of pedestrian evacuation. *Physica A*, 354:606–618, 2005.
- [13] A. Schadschneider, W. Klingsch, H. Kluepfel, T. Kretz, C. Rogsch, and A. Seyfried. *Encyclopedia of Complexity and System Science*, volume 3, chapter Evacuation Dynamics: Empirical Results, Modeling and Applications, page 3142. Springer, 2009. preprint via arXiv:0802.1620.
- [14] A. Seyfried, M. Boltes, J. Kähler, W. Klingsch, A. Portz, A. Schadschneider, B. Steffen, and A. Winkens. Enhanced empirical data for the fundamental diagram and the flow through bottlenecks. In *Pedestrian and Evacuation Dynamics 2008*. Springer, 2009. arXiv:0810.1945, in print.

LITERATURVERZEICHNIS

- [15] A. Seyfried, T. Rupprecht, O. Passon, B. Steffen, W. Klingsch, and M. Boltes. New insights into pedestrian flow through bottlenecks. *Transportation Science*, may 2009. arXiv:physics/0702004.
- [16] A. Seyfried, B. Steffen, and T. Lippert. Basics of modelling the pedestrian flow. *Physica A*, 368:232–238, 2006.
- [17] B. Steffen and A. Seyfried. The repulsive force in continuous space models of pedestrian movement. 2008. arXiv:0803.1319v1.
- [18] G. Sutmann and V. Stegailov. Optimization of neighbor list techniques in liquid matter simulations. *Journal of Molecular Liquids*, 125(2-3):197–203, 125 2006.
- [19] L. Verlet. Computer Experiments on Classical Fluids. I. Thermodynamical Properties of Lennard-Jones Molecules. *Phys. Rev.*, 159(1):98–103, 1967.
- [20] Z. Yao, J.-S. Wang, G.-R. Liu, and C. Min. Improved neighbor list algorithm in molecular simulations using cell decomposition and data sorting method. *Computer Physics Communications*, 161(1-2):27–35, aug 2004.
- [21] W. J. Yu, L. Chen, R. Dong, and S. Dai. Centrifugal force model for pedestrian dynamics. *Phys. Rev. E*, 72(2):026112, aug 2005.

Jül-4308
September 2009
ISSN 0944-2952